University of Leipzig
Faculty of Medicine

Institute for Medical Informatics, Statistics and Epidemiology
(IMISE)

Integration of a Medical Documentation and Image Archiving System within Hospital Information Systems with the Utilisation of a Web-Based User Interface

Diploma Thesis

Leipzig, 23rd May 2006

written by: Hans-Holger Schmuhl
DOB: 24th February 1980
Instructor: Prof. Dr. Alfred Winter (IMISE)
Tutors: Dr. Thomas Wendt (IMISE)
Bernd Nuber (ViewPoint)
Bernhard Hornung (ViewPoint)

Acknowledgement

To this diploma thesis contributed quite a lot of people.

First of all, I like to thank my parents to whom I owe that I got so far at all. They supported me wherever they could and especially encouraged my interest in sciences.

Then, I like to thank ViewPoint in general. Not only for offering me this thesis and supporting me in writing it. They are the ones who finally brought me to study Medical Informatics. In special, I like to thank Bernhard Hornung and Bernd Nuber who looked after me during this thesis. Furthermore, I like to thank all my colleagues at ViewPoint who instantly and willingly answered my questions in detail.

Moreover, I like to thank Dr. Thomas Wendt and Prof. Dr. Alfred Winter at IMISE for enabling the cooperation with ViewPoint and especially for supporting me in writing this thesis.

In addition, I like to thank Svein Fimreite, John Stepleton and Christian Harböck for providing me a comprehensive insight in their products and thoughts.

Furthermore, I like to thank Ike Biedermann for the proofreading of my English.

Finally, I like to thank Susann for her everlasting patience and support.

Holger Schmuhl, May 2006

Contents

Contents v

List of Figures xi

List of Tables xiii

1. Introduction 1

1.1. Outline . . . 1

1.2. Subject and Motivation . . . 1

1.2.1. Subject . . . 1

1.2.2. Problem Definition . . . 2

1.2.3. Motivation . . . 4

1.3. Problems . . . 5

1.4. Objectives . . . 5

1.5. Question- and Taskdefinition . . . 6

1.6. Stakeholders . . . 7

2. Basic Principles 9

2.1. Outline . . . 9

2.2. Disambiguation . . . 9

2.2.1. General Terms . . . 9

2.2.2. Integration Types . . . 10

2.2.3. Relevant Communication Standards in Health Care . . . 10

2.2.4. Relevant Application Components in Hospital Information Systems . 11

2.3. Requirements Engineering . . . 12

2.3.1. Unified Modelling Language . . . 15

2.3.2. 3LGM^2 Meta Model . . . 16

2.3.2.1. Introduction . . . 16

2.3.2.2. Layers and Their Relationship 17
2.3.2.3. Reference Model for the Domain Layer of HIS 18
2.3.2.4. 3LGM^2 Modelling Tool 19
2.4. Methods of Information Procurement 19
2.4.1. Interview 19
2.4.2. Questionnaire 19
2.4.2.1. Questionnaire “Features of a Web-based Interface to VP” . 20
2.4.2.2. Questionnaire “Product Analysis” 21
2.4.3. Analysis of Existing Data 21
2.5. Product Analysis 22
2.6. Native Client versus Web Client 22
2.6.1. Native Client 23
2.6.2. Web Client 24
2.6.3. Comparison 26
2.7. Terminal Server 26
2.8. Clinical Context Management 28

3. Requirements Analysis **31**
3.1. Outline 31
3.2. Current Functionality of the Native ViewPoint Client 31
3.2.1. Hardware and Software Requirements of a Client 31
3.2.2. Installation Procedure 32
3.2.3. General Description 32
3.2.4. Patient Management Module 33
3.2.5. Patient Scheduling Module 35
3.2.6. Statistics Module 36
3.2.7. Archive Module 37
3.2.8. Examination Module 37
3.2.8.1. Case and Examination Management Submodule 37
3.2.8.2. Findings System 39
3.2.8.3. Image Gallery Submodule 43
3.2.8.4. Printing and Faxing Submodule 45
3.2.8.5. Rule-Checking Submodule 45

3.2.9. System Administration Module (VPAdmin) 46
3.3. Current State of Integration 49
3.3.1. Identification of Available Interfaces 49
3.3.1.1. HL7, HCM, Clinicom, BDT and Other (computer-based) . 49
3.3.1.2. DICOM (computer-based) 50
3.3.1.3. CORBA (computer-based) 52
3.3.1.4. Analog Image / Video Capture (computer-based) 52
3.3.1.5. US Measurement Data (computer-based) 52
3.3.1.6. Trium CTG Online Monitoring (computer-based) 53
3.3.1.7. Chip Card Reader (computer-based) 53
3.3.1.8. Printing and Faxing Interface (paper-based) 53
3.3.1.9. Scanning Interface (paper-based) 53
3.3.1.10. Other 53
3.3.2. Model of Current State of Integration 54
3.4. Weak Point Identification 54
3.4.1. Weak Points of the Native VP Client 55
3.4.2. Weak Points in Respect to Integration Potentials of the Native VP Client 56
3.4.2.1. Data Integration 56
3.4.2.2. Access Integration 57
3.4.2.3. Presentation Integration 58
3.4.2.4. Contextual Integration 59
3.4.3. Weak Points of a Web Client 59
3.4.3.1. Weak Points of a Web Client in General 59
3.4.3.2. Points to Be Considered for a Web Client as an Alternative UI to VP 61

4. Definition of Requirements **63**
4.1. Outline 63
4.2. Requirement Definition 63
4.3. Future State of Integration 66

5. Specification of Requirements **69**
5.1. Outline 69

5.2. Research of Open Issues . . . 69
5.2.1. Quantification of the Features Offered by the Native Client . . . 69
5.2.2. Identification of Application Components where VP is Needed . . . 71
5.2.3. Identification of Platforms and Browsers which Should Be Supported by the Web Client . . . 72
5.2.4. Identification of Required Web Browser Extensions . . . 74
5.2.5. Interaction with Other Application Components . . . 77
5.3. Requirement Specification . . . 78

6. Technology Comparison **85**
6.1. Outline . . . 85
6.2. Product Analysis and Comparison . . . 85
6.2.1. Euroking – E3 . . . 85
6.2.1.1. General Description . . . 85
6.2.1.2. Technical Aspects . . . 86
6.2.2. GE Vingmed Ultrasound – WebEx . . . 88
6.2.2.1. General Description . . . 89
6.2.2.2. Technical Aspects . . . 90
6.2.3. Trium CTG Online . . . 91
6.2.3.1. General Description . . . 92
6.2.3.2. Technical Aspects . . . 93
6.2.4. Definition of Comparison Criteria . . . 93
6.2.5. Product Comparison . . . 95
6.2.6. Justification of Requirements . . . 96
6.3. Terminal Client as Alternative to a Web client . . . 98
6.3.1. Terminal Server and VP . . . 98
6.3.2. Web Client vs. Terminal Client . . . 101

7. Conclusion **105**
7.1. Outline . . . 105
7.2. General Discussion . . . 105
7.3. Achievement of Objectives . . . 106
7.3.1. Objective 1 . . . 106
7.3.2. Objective 2 . . . 107

7.3.3. Objective 3 110
7.4. Prospect 111

Bibliography **113**

Index **116**

A. Appendix **121**

B. Declaration **139**

List of Figures

2.1. Spiral model of Boehm, taken from [Som95]. 13

2.2. Requirements engineering process according to [Som95]. 14

2.3. Client-server architecture, taken from [Tan03]. 23

2.4. Web browser plug-In and helper application, taken from [Tan03]. 25

2.5. Terminal server with one local and three remotely connected users, modified version of a figure taken from [Mat04]. 27

2.6. Overview of the components of the CCOW CMA, taken from [Sel01]. 29

3.1. Patient Management Module – UML use case model of the management of patient records, modified version of the model taken from [Seh05]. 34

3.2. Patient Scheduling Module – UML use case model of the management of appointments, modified version of the model taken from [Seh05]) 36

3.3. Simplified UML data model of ViewPoint 38

3.4. Findings System, Prenatal Reporting – screenshot of the biometry submask. 40

3.5. Findings System, ViewPoint Report – screenshot of the report modification mask. 41

3.6. Findings System, ViewPoint Report – UML use case model of the report handling, modified version of the model taken from [Seh05]. 42

3.7. Findings System, Archiving Process – UML use case model, modified version of the model taken from [Seh05] 43

3.8. Image Gallery Submodule – UML use cases of image data handling, modified version of the model taken from [Seh05] 44

3.9. Printing and Faxing Submodule – First side of an obstetrical ultrasound report. 47

3.10. Printing and Faxing Submodule – Second side of an obstetrical ultrasound report. 48

3.11. Logical and physical tool layer of the 3LGM^2 model showing the two alternatives of the DICOM storage concept of ViewPoint. 51

3.12. Logical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system. 57

3.13. Physical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system. 58

4.1. Logical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system. 67

4.2. Physical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system. 68

5.1. Sample screenshot of Java Applet RemoteEye of NeoLogica s.r.l., taken from [Neo06]. 76

6.1. Screenshot showing the patient selection screen of E3. 86

6.2. Screenshot showing the pregnancy summary of E3. 87

6.3. Screenshot showing the search screen of WebEx. 88

6.4. Screenshot showing the image gallery screen of WebEx. 89

6.5. Screenshot showing the overview screen of Trium CTG Online. 91

6.6. Screenshot showing the archive screen of Trium CTG Online. 92

6.7. Screenshot of ViewPoint's Native client running in the Microsoft Remote Desktop Web Connection. 100

A.1. Logical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system. 123

A.2. Physical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system. 124

A.3. Logical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system. 125

A.4. Physical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system. 126

List of Tables

2.1. 3LGM^2 meta model – graphical representation of elements at the domain layer. 17

2.2. 3LGM^2 meta model – graphical representation of elements at the logical tool layer. 18

2.3. Comparison of Native client and Web client architectures. 26

3.1. Overview of the interfaces of ViewPoint to CDMS and PMS. 50

5.1. Quantification of features which are offered by the Native client and shall be offered by the Web client. 70

5.2. Comparison of statistics taken from www.TheCounter.com [The06] and www.W3Schools.com [W3S06] in march 2006. 73

5.3. Significance of supporting referring OS and Web browser. 74

5.4. Web browser and OSs which should be supported by the Web client. 74

6.1. Comparison of E3, WebEx and Trium CTG Online. 95

6.2. Comparison of Web Client vs. Terminal Client 101

7.1. Short Overview of the requirements for a Web Client 108

A.1. Enterprise functions originating from the reference model of the domain layer proposed by Hübner-Bloder et al. [HBABW05] which are supported by the corresponding modules of ViewPoint. 121

1. Introduction

1.1. Outline

The aim of this chapter is to give a general description of the subject of this thesis. Furthermore, it will give reasons for why it is relevant to deal with this topic. Both will be described in section 1.2. Section 1.3 will list problems which additionally motivate this thesis. Section 1.4 will define clear objectives which should be achieved in the course of this thesis. Additionally. Section 1.5 will give a question and task definition to define a rough framework to successfully complete the thesis. Finally, the stakeholders of this thesis will be listed in section 1.6 to highlight their interests in it.

The succeeding chapter will give a description of basic principles which form the basis of this thesis.

1.2. Subject and Motivation

1.2.1. Subject

The ViewPoint (VP) image archiving and reporting system enables digital acquisition, communication and archiving of single image and multi frame data[1] and the recording of structured findings in areas of gynaecology, obstetrics, internal medicine and general ultrasound. VP is made and supported by ViewPoint Bildverarbeitung GmbH - GE Healthcare Technologies (VP-GE).

Concerning the offered functionality it can be seen as Medical Documentation System (MDS) and Picture Archiving and Communication System (PACS) inside the classes of application components [HAWB04].

For a start, a description of the scenario in which VP is used will be given to clarify the subject of this thesis.

The application of VP is initiated by the placement of an order at the general practitioners office, ward or outpatient unit. At this point, an examination in a diagnostic service unit is requested by the physician responsible. The order is sent in electronic or paper form to the executing department. There a new patient record is created or if he/she was already at the department, the existing patient record is selected in VP.

All data emerging in the course of examination is recorded in VP. This comprises findings which are entered in structured findings masks. In these findings masks the user can choose from predefined clauses or enter free text. The structured findings are based upon

[1] By reasons of simplicity single images (also called still pictures) and multi frames (also called video sequences or loops) will be denoted by the term image data if no further distinction is required.

guidelines issued by the respective professional societies. Beside findings, VP offers also functionality to record image data like single images or sequences from modalities like ultrasound machines or endoscopy devices.

Based on the findings, the physician can enter diagnoses and therapeutic recommendations in VP.

At the end of the examination and assessment process, the health care professional can print out different types of forms, like for example a referral or discharge letter, an internal report etc. and also include images or graphs. Moreover, the examination data can be communicated electronically for billing purposes. The report text is automatically created from the content of the masks. However, the health care professional has the possibility to modify this text.

Beside that, VP offers functions for patient scheduling, execution of basic statistical queries and image post-processing.

The application of VP ends when the report about the examination is sent to the requesting physician or unit and further treatment of the patient is based on these results.

From a technical point of view, the system is based on the client-server architecture. The server stores the central database and image archive. Furthermore, it offers functions for electronic data exchange and backup. The clients differ in their functionality depending on the license purchased. Minimal functionality offer so called viewing stations, with which findings and image data can only be viewed without modifying capabilities. Maximal functionality offer so called image acquisition stations, with which the capturing (analog modalities)/reception (from DICOM modalities), editing and archiving of image data is feasible in addition to the possibility to enter and modify findings.

Apart from medical practices, the system is mainly utilised in hospitals. There, it is part of the Hospital Information System (HIS) and tightly coupled with the Patient Management System (PMS). In the scope of this diploma thesis, only the case in which VP is deployed inside a hospital will be considered.

As more and more hospitals change to a paperless HIS, it is indispensable to facilitate access to image data and findings in the VP system from every workstation at every time.

At the moment, this is only possible if a VP client is installed on the accessing workstation.

1.2.2. Problem Definition

The fact that VP needs to be installed on a workstation to be able to view data archived in the VP system may cause some unpleasant concomitants.

As the main aim of nearly every hospital is to come as close as possible to an integrated HIS, the installation of a VP client on an existing workstation and the introduction of a new application component in general implies a tight coupling with other application components to realise as much integration types [HAWB04, 127-129] as possible.

To start with, Data Integration in short means data has to be recorded, modified or deleted just once and is available wherever it is needed [HAWB04, 127]. As VP also handles data which is redundantly stored in other application components, for example

patient demographics [2], diagnoses, procedures, findings and image data, it is inescapable to offer interfaces for electronic data exchange with other application components to enable data integration. At the moment VP offers various interfaces for the electronic exchange of orders, patient demographics, diagnoses, procedures, findings and image data like Health Level 7 (HL7), BDT and DICOM.

Access Integration in short means that the related application component is available wherever it is needed [HAWB04, 128]. In case of VP it is common to only install the system in the diagnostic service unit which examines the patient and acquires the data, for example internal medicine or general ultrasound. Normally no client is available at the ward or other assigning departments which are probably also interested in the data. For physicians or other health care professionals this circumstance implies that they have to find a workstation on which the VP client is installed or to print out the relevant data to access the needed information at their place. However, this fact contradicts with the original idea of a paperless and integrated HIS and leads to media cracks.

In order to guarantee Presentation Integration, data and the User Interface (UI) of different application components should be represented in unified ways [HAWB04, 128]. At present this scenario is far from reality. Even if application components are produced by the same manufacturer, they mostly come with different UI and data representations, as there exist quite different and partly controversial opinions on software ergonomics. This fact also applies to VP.

Contextual Integration means that the context is preserved while switching between application components on one workstation [HAWB04, 128]. Although a standard for context integration published by the Clinical Context Object Workgroup (CCOW) exists, it is implemented by very few manufacturers and not yet supported by VP.

Now, to focus on a more technical view, VP needs an adequate soft- and hardware equipment to enable a smooth, trouble-free and user friendly programme execution. Although the installation and maintenance of a VP client is comparatively simple, every extra client is connected with additional expenses. Especially when implementing an integrated HIS, the number of potential clients will compulsorily increase.

For the IT department that means that every extra client is a new software component and in case the client is installed on a new computer also a new hardware component which is placed in their area of responsibility. The client has to be maintained and its users have to be adequately supported. If the hard- and/or software equipment is not sufficient, it has to be upgraded which beside extra costs may cause compatibility problems with other application components. Furthermore, it is clear that with a growing number of workstations running VP the probability for inevitable hard- or software errors will also increase accordingly.

As already mentioned in the section above about access integration, the non-existence of access integration compulsorily leads to media cracks, in which the storage media is changed in the course of transcription [HAWB04, 160], for example when an internal report is printed out at the diagnostic service unit to have patient data available at the ward or when a CD-ROM is burned for the assigning general practitioner so that she/he

[2] The term patient demographics denotes data which describes basic properties of a patient. It stays in the course of time stable. [LGH+03] Patient demographics means in the context of this thesis patients name, address, assigning physician and health insurance information. It is also called demographic patient data.

has the image data in high quality and with the ability to post-process it at her/his practice. Beside the duplication of data which causes the commonly known problems of data redundancy, media cracks may also lead to the inaccessibility of data, as the reading device will not be available in future or is not available at all places where the media should be read.

Speaking in the context of an integrated health care region, where remote sites are integrated in the HIS as if they are situated within the hospital, integration criteria have to be realised under circumstances which are inflicted by the special characteristics of Wide Area Network (WAN) connections. VP system is not originally designed for teleconsulting purposes, as the client needs a permanent high bandwidth connection to access the VP server. Therefore, VP is not able to support the integration of health care regions adequately.

1.2.3. Motivation

The listed problems can be eliminated or at least reduced to a minimum by a Web-based UI to the VP system.

With the utilisation of a Web-based UI, the user is able to access VP by using a standard Web browser. The functional range, which is made available via the Web-based UI has yet to be determined. Irrespective of the functionality, this access method results in a set of significant advantages:

Data access via Web browser is platform independent, as some Web browsers are available for nearly every OS (for example Firefox). The only precondition is that the accessing workstation needs to be networked with the VP server.

Considering those two points and the aspect that nearly every workstation within the hospital is interconnected via the hospital's Intranet, a Web-based UI enables access integration for VP. The degree of access integration will be determined by the functionality which is offered by the Web-based UI.

Today the usage of a Web browser is familiar to almost everybody, as the distribution of the Internet has reached nearly every household. So no special training is needed to learn the basic usage of Web-based UI. Presentation integration is certainly not guaranteed by this solution, but as more and more vendors offer Web-based UI for their software product and the handling of a Web browser is at least uniform, it is one step in the right direction to presentation integration.

The level of data integration will not be changed by a Web-based UI, as it will not offer any new interfaces for electronic data exchange with other application components. The Web-based UI will just be an add-on for the VP server which is responsible for electronic data exchange.

For context integration a Web-based solution may be the ideal solution. The exact realisation of context integration is at this point still in question. Beside implementing the CCOW standard, which is also available for Web-based application components, it is also conceivable to open a certain patient record and to send the required login information via a specific Uniform Resource Locator (URL) like on certain sites on the Internet.

Furthermore, a Web browser compared to a VP client needs a minimum of soft- and

hardware equipment and is a built-in application component of nearly every OS these days. In case it is not, it can be simply installed without causing any compatibility problems with other software components.

As already mentioned, the Web-based UI is accessible from every workstation which is connected to the VP server. So with the utilisation of the Internet or a dedicated (dial-in) network connection, data access is also possible from a remote site, as it may be the case that a physician wants to complete her/his work from home or that an assigning physician wants to review the results of a requested examination. By using standard WWW-techniques, the connection has not to be of permanent type and can also be of lower bandwidth.

Although the Web-based UI might at first glance seem like an ideal solution to most listed problems, also Terminal server technology should also be briefly examined as an alternative as it offers similar advantages.

1.3. Problems

Problem 1: Until now the access from many workstations to the VP system is highly complex.

Problem 2: VP is a highly specific medical documentation system, so that the means of for example HL7 are not sufficient to adequately integrate VP within the HIS.

Problem 3: Until now no Web-based UI for VP exists.

Problem 4: Until now at VP-GE no sound knowledge of the requirements and expectations of a Web-based UI exists.

1.4. Objectives

Objective 1: Determine which requirements are not met by the Native client of VP and the VP system in general for an adequate integration within the HIS.

Objective 2: Establish a recommendation for what technology should be used to implement a Web client.

With the Web client access to VP should be possible from every workstation which is networked with the VP server, without any additional application components, and by using a standard Web browser.

Limitations: A selection of software products with a Web-based UI should be examined and described to also consider existing solutions during the creation of the concept. Selection is made in accordance with VP-GE and Institute for Medical Informatics, Statistics and Epidemiology (IMISE).

Terminal server technology should also be examined as an alternative technique to the Web-based access to VP.

Objective 3: The degree of integration of VP in the HIS should be analysed and modelled by using the 3LGM^2 meta model. The current integration degree with the Native

client and the future integration degree with the additional Web client should be examined.

1.5. Question- and Taskdefinition

Question 1: Which requirements should be fulfilled by the Web-based UI?

Task 1.1: Identification of the available functions at a VP client.

Task 1.2: Determination of missing functions and additional functions which should be made available, especially enabled by the special characteristics of the Web environment (for example remote access etc.).

Task 1.3: Weighting/quantisation of listed functions to have a sound idea of which functions are essential to be adopted by the Web-based interface and which are optional. Weighting/quantisation is based upon the input from the sales, service and development departments of VP-GE.

Result: system requirement specification
Aspired completion: 1.12.2005

Question 2: How are certain Web-based UI of medical documentation systems which are available at the market realised?

Task 2.1: Identification of other application components already offering a Web-based UI.

Task 2.2: Analysis and evaluation of the application components found.

Task 2.3: Review and probable modification of the system requirement specification due to new ideas or concepts taken from other application components.

Results: reviewed system requirement specification, overview of the technology which is used by partners/opponents to realise a Web-based UI
Aspired completion: 31.01.2006

Question 3: How is VP integrated in the HIS at the moment? What will change with the utilisation of a Web-based UI?

Task 3.1: Analysis of the current state of integration without a Web-based UI.

Task 3.2: Analysis of the future state of integration with a Web-based UI.

Task 3.3: Modelling of both states using the $3LGM^2$ Meta model.

Result: descriptions and models of the current and future state of integration
Aspired completion: 10.02.2006

Question 4: To what degree is Terminal server technology an alternative to the Web-based UI of VP? What are its advantages and disadvantages?

Task 4.1: Introduction to Terminal server technology.

Task 4.2: Examination of how Terminal server technology can support the integration of VP in the HIS.

Result: brief idea of whether Terminal server technology can be a serious alternative to the Web-based approach.
Aspired completion: 24.02.2006

Question 5: What are the pros and cons of the different methods of resolution? What methods should be chosen?

Task 5.1: Setup of different concepts realising the system requirements by Web-based and alternative (non Web-based) technologies.

Task 5.2: Comparison of the different concepts (incl. Terminal Server/Thin Client technology).

Task 5.3: Identification of the concept which suits best.

Result: technology comparison, recommendation of the concept with best benefits
Aspired completion: 06.03.2006

Question 6: Is the chosen concept adequate to fulfil the given requirements?

Task 6.1: Determine the requirements which cannot be fulfilled by the chosen concept.

Result: basis for further considerations
Aspired completion: 31.03.2006

1.6. Stakeholders

This section is intended to clearly state the stakeholders of this diploma thesis and their interests in it.

ViewPoint Bildverarbeitung Gmbh - GE Healthcare: Their interest on this diploma thesis is to obtain a requirements specification of a Web-based UI to VP. Furthermore they want to gain an overview of which technology is used by other vendors and which technology is the most suitable one for their purposes.

ViewPoint Bildverarbeitung GmbH
GE Healthcare Technologies
Argelsrieder Feld 12
D-82234 Wessling
`http://www.gehealthcare.com/viewpoint`

Institute for Medical Informatics, Statistics and Epidemiology: Their interest is to obtain an impression on how a Web-based interface can change the degree of integration of a medical documentation and image archiving system.

Institut für Medizinische Informatik, Statistik und Epidemiologie
Universität Leipzig
Härtelstraße 16-18
D-04107 Leipzig
`http://www.imise.uni-leipzig.de`

Holger Schmuhl: He is the author of this thesis. His interest is to successfully complete it by meeting the expectations of both parties.
eMail: holger.schmuhl@web.de

2. Basic Principles

2.1. Outline

In the preceding chapter a general introduction to the topic of this diploma thesis has been given.

The aim of this chapter is to give a sound description of principles which form the basis of this thesis. The reader should thereby gain accurate knowledge of the used terms, methods and technology so that she/he is able to follow the course of the thesis. First, a disambiguation will be given in section 2.2 to prevent misunderstandings. Section 2.3 will focus on requirements engineering which is one central objective of this thesis. Section 2.4 will show the methods which have been used to gather the required information. Following in section 2.5, the general idea behind product analysis will be highlighted. Afterwards, in section 2.6 the concepts behind a Native and a Web client will be introduced and compared to each other. Furthermore, the concept of a Terminal client will be generally introduced in section 2.7. This chapter ends with giving an overview of CCOW in section 2.8.

The succeeding chapter will then start to deal with the actual subject of this thesis by analysing the requirements for a Web-based UI.

2.2. Disambiguation

The intention of this paragraph is to clearly state and highlight the meaning of certain terms. Thereby, misunderstandings should be prevented as there are quite different terminologies in use. In general the terminology proposed by IMISE will be used. Most parts have been published in [HAWB04] and have been defined while creating the 3LGM^2 Meta model (see section 2.3.2 for details).

2.2.1. General Terms

Hospital Information System (HIS): By the term Hospital Information System most people mean what is actually denoted in the terminology by PMS or sometimes Clinical Documentation and Management System (CDMS). Correctly a HIS is the socio-technical subsystem of a hospital that comprises all information processing functions and the human or technical actors in their information processing role. [HAWB04]

Image Data: The term image data denotes single images (also called still pictures) and multi frames (also called video sequences or loops) if no further distinction is required.

Platform: There are different definitions of the term Platform. In this thesis the term Platform denotes a hardware architecture, like the Intel x86 hardware architecture together with the used Operating System (OS), like Microsoft Windows.

Web client versus Web-based UI: In this thesis the terms Web client and Web-based UI are synonymously used.

Authorisation versus Authentication: Authorisation denotes the process in which an actor/party (like for example an user) is granted access to certain data. The corresponding data is protected in such a way that only selected users who are legitimate can access it. Authentication denotes the process in which it is verified that the actor/party truly is the actor/party she/he/it claims to be and not an imposter.

Terminal server: In this thesis the term Terminal server denotes not only the server itself that provides certain services in a client-server architecture. In addition Terminal server denotes the technology in general which is used in a Terminal client-server architecture.

2.2.2. Integration Types

Integration: Integration is a union of parts making a whole, which as opposed to its parts, displays a new quality. An integrated HIS is a union that represents more than just a set of independent components.[HAWB04]

Data Integration: "Data integration is guaranteed in a hospital information system when data that have been recorded are available wherever they are needed, without having to be reentered. Thus, each data item needs to be recorded, changed, deleted, or otherwise edited just once even if it is used in several application components. Data integration is a prerequisite for the multiple use of data." [HAWB04]

Access Integration: "Access integration is guaranteed when the application components needed for the completion of a certain task can be used where they are needed." [HAWB04]

Presentation Integration: "Presentation integration is guaranteed when different application components represent data as well as user interfaces in a unified way." [HAWB04]

Contextual Integration: "Contextual integration means that the context is preserved when the application component is changed." [HAWB04]

2.2.3. Relevant Communication Standards in Health Care

Health Insurance Portability and Accountability Act (HIPAA): The Health Insurance Portability and Accountability Act is a national standardisation approach for electronic health care transactions and national identifiers for providers, health plans, and employers in the USA. It also addresses the security and privacy of health data. It is issued by the United States Department of Health and Human Services. Official Web sites: `http://www.hhs.gov/ocr/hipaa` and `http://www.cms.hhs.gov/HIPAAGenInfo`

BDT: The term BDT (short form of the German term "Behandlungsdatenträger") denotes a standard which is used at general practitioner's practices to exchange administrative and medical information between different electronic practice management systems. The standard is issued by the Zentralinstitut für die kassenärztliche Versorgung Bundesrepublik Deutschland. Official Web site: `http://www.zi-berlin.de`

Clinical Context Object Workgroup (CCOW): The term Clinical Context Object Workgroup denotes a standard used to automatically synchronise the patient and user context between different application components. It is issued by the Clinical Context Object Workgroup of the HL7 Committee. The Clinical Context Object Workgroup is sometimes also denoted by the term CCOW. See section 2.8 for details. Official Web site: `http://www.hl7.org.au/CCOW.htm`

Digital Imaging and Communications in Medicine (DICOM): The term Digital Imaging and Communications in Medicine denotes a standard used to handle, transfer and store medical images and related information. It is issued by the National Electrical Manufacturers Association (NEMA). Official Web site: `http://medical.nema.org`

Health Level Seven (HL7): The term HL7 denotes a set of standards for information interchange in health care. The standards are issued by the Health Level Seven Inc. which is also called HL7 Committee. Official Web site: `http://www.hl7.org`

HCM: HCM is a proprietary standard of SAP AG for information interchange in health care. Official Web site for more information: `http://www.sap.com`

Clinicom: Clinicom is a proprietary standard of Siemens Medical Solutions for information interchange in health care. Official Web site for more information: `http://www.medical.siemens.com`

SAP Remote Function Call (SAP-RFC): SAP Remote Function Call is a proprietary standard of SAP AG for information interchange in general. Official Web site for more information: `http://www.sap.com`

2.2.4. Relevant Application Components in Hospital Information Systems

Clinical Documentation and Management System (CDMS): The Clinical Documentation and Management System is a computer-based application component that supports documentation and management tasks clinic wide.

Medical Documentation System (MDS): The Medical Documentation System is a computer-based application component that supports specific documentation tasks (like patient history, report writing etc.) in different medical fields (like obstetrics, gynaecology, internal medicine etc.) [HAWB04].

Picture Archiving and Communication System (PACS): The Picture Archiving and Communication System is a computer-based application component that supports the storage, management and presentation of digital Image Data. Moreover it offers means for transferring the Image Data from the storage media to the attached workstations. It may offer additional functions for image processing [HAWB04].

Patient Data Management System (PDMS): The Patient Data Management System is a computer-based application component that is specialised in automatically monitoring, storage and presentation of patient related clinical data [HAWB04].

Patient Management System (PMS): The Patient Management System "is a computer-based application component that supports patient admission, discharge and transfer" [HAWB04].

Radiology Information System (RIS): The Radiology Information System is a computer-based application component that "offers functions for appointment scheduling, organisation of examinations and staff [...], provision of patient data and examination parameters, and the creation of reports" [HAWB04].

2.3. Requirements Engineering

As already stated in the introduction, the aim of this thesis is not to develop a running system, but to create a concept or more precisely a recommendation of what technology should be used.

From a project perspective, this thesis has some special characteristics compared to a standard software development project, in which a software is created from scratch. Following these will be listed,

- Subject is to develop a concept (technical recommendation) to enable Web-based access to an already existing product, leaving the actual functions of the product untouched.
- An requirements specification already exists, defining the actual functions of the product. It does not contain any requirements to address the demand for offering a Web client.

This implies that this project does not follow the course of a certain software (development) process, which starts from scratch and ends with the final product being released. However, according to the four fundamental process activities which are common to all software processes (software specification, development, validation and evolution) described in [Som95], it passes through the first phase in which the functions of the software and constrains on its operation are defined.

Nevertheless it was initially intended to use parts of an existing, well-defined and described software process model to have a sound guideline for conducting all required activities in the right order and to the right extend. Thus, after gaining a general idea of the process families [BK02] and certain representatives like the spiral model of Boehm [Bal98] or the Rational Unified Process [KK03], it came up that following a specific process model would not substantially contribute to the goal of this thesis. It would rather complicate the general procedure without offering any compensatory benefits.

In most process models, activities of the software specification are closely related to activities of other phases like implementation or validation. This is, for example, characteristic for iterative process models, in which the phases are manifoldly passed through. Moreover, the results produced and experiences made in other phases have a strong influence on the further development and successful completion of the software specification. As the other phases will not, however, be passed through in this thesis and would therefore not give any input to the specification phase, it was seen as more reasonable to refer to literature which is describing requirements engineering and its methodology in a more general way.

According to Davis et al. requirements engineering "applies proven principles, techniques, languages, and tools to help analysts understand a problem or describe a potential product's external behaviour" [DH94].

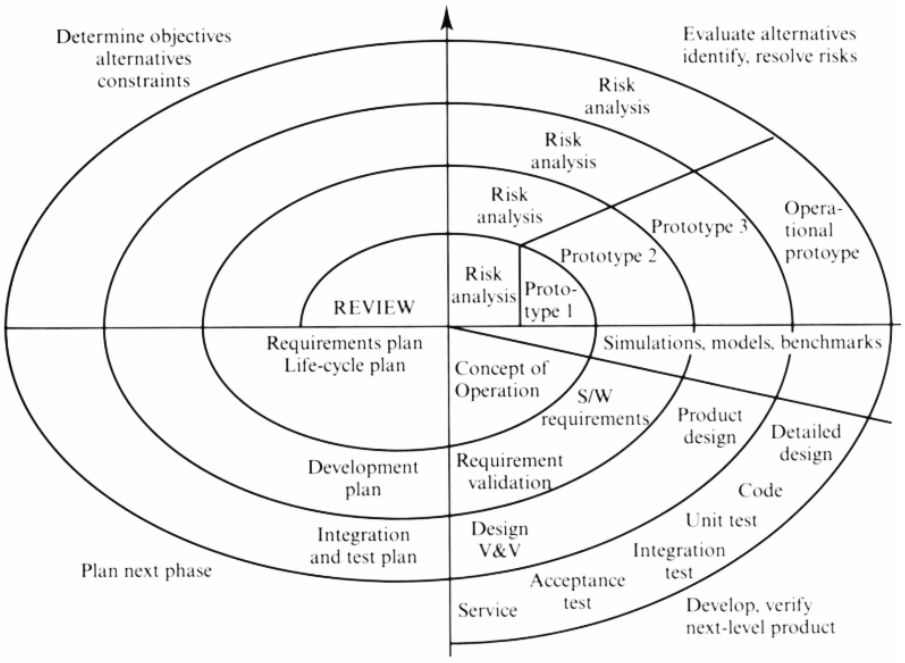


Figure 2.1.: Spiral model of Boehm, taken from [Som95].

In general [Som95] was taken as reference and guideline to come up with a requirements specification. Sommerville described the requirements engineering process as a process consisting of four principle stages, which will subsequently be outlined in short. In addition, it will be referenced how and where it was realised in this thesis:

1. *Feasibility study* according to [Som95]:

   The intention of this stage is to estimate whether the user needs may be satisfied using current software and hardware technologies. The focus is on economical aspects like cost-effectiveness and budgetary constrains. The completion of this stage should result in a feasibility report.

   In this thesis a feasibility study will be completely omitted as economical analysis is not subject of this thesis and the decision of the actual implementation is at the hands of VP-GE.

2. *Requirements analysis* according to [Som95]:

   - Requirement analysis is the process of deriving the system requirements through various methods. [Som95]
   - Core part of this stage is to understand the problem domain to develop a sound knowledge of it. [Som95]
   - Requirement analysis may involve the development of system models. [Som95]

   In order to fully understand the problem domain, the functions of the current VP version will be presented in section 3.2 and the current state of integration will be analysed in section 3.3. Based on these information the current weak points with

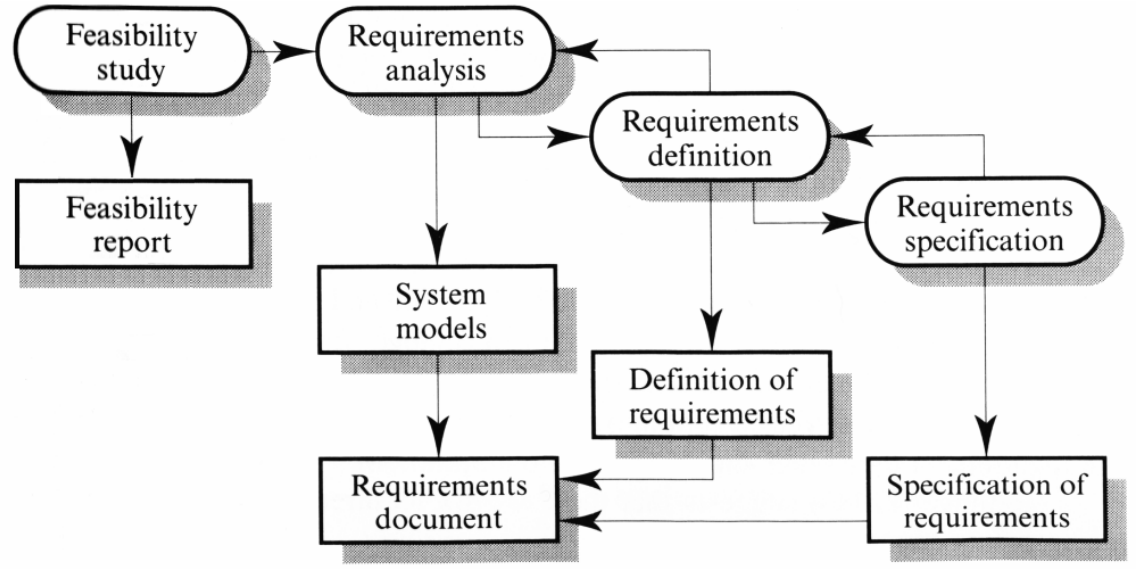


Figure 2.2.: Requirements engineering process according to [Som95].

prospect to integration will be identified in section 3.4 and the requirements will be derived from them.

In order to illustrate complex functions and the involved actors, Universal Modelling Language (UML) Use Case Meta model will be used. In order to analyse the state of integration, the 3LGM^2 Meta model will be applied.

3. *Requirements definition* according to [Som95]:

   - Requirements definition is the activity of translating the information gathered during the analysis activity into a document that defines a set of requirements. [Som95]
   - It should be written in an abstraction level that can be understood by an end-user and the system customer. [Som95]

   Sommerville distinguished between two types of requirements: *Functional* requirements (statements on services the system should provide) and *Non-Functional* requirements (constraints on the services or functions offered by the system). This distinction will not be made in this thesis as it is seen as more significant to arrange the requirements topically. The requirements definition will be given in section 4.2.

   Additionally, Sommerville defined different types of major problems of requirements definitions. They will be subsequently listed in conjunction with which was done to cope with them:

   - *Lack of clarity*: "It is very difficult to use language in a precise and unambiguous way without making the document wordy and difficult to read." [Som95]

     In order to achieve as much clarity as possible it will be focused on making simple, clear and unambiguous statements without using fill words or other stylistic means to make the text read smoother.

- *Requirements confusion*: "Functional requirements, non-functional requirements, system goals and design information may not be clearly distinguished." [Som95]
  As stated above, it will purposely not be distinguished between the different types of requirements in this thesis. It is seen as more crucial to group the requirements according to the general topic they aim at like data security, quality of supported networks or Platform, OS version and Web browser constrains.
- *Requirements amalgamation*: "Several different requirements may be expressed together as a single requirement." [Som95]
  During requirement definition it was tried to densify the requirements as much as possible.

4. *Requirements specification* according to [Som95]:

   - Requirements specification is a detailed and precise description of the system requirements. [Som95]

   On account of the following reasons, Sommerville et al. warned of using natural language for requirements specification: Natural language understanding is reader and writer dependent which may lead to misunderstandings. Furthermore natural language specification is over-flexible so that the same thing can be expressed in different ways. This may lead to problems in finding out whether different requirements are actually the same or distinct. Moreover, requirements are not effectively partitioned by the language itself and it is thus difficult to find related requirements.

   Still, the use of natural language is seen as more appropriate in this thesis to specify the requirements. Other forms like graphical notations, mathematical specifications or requirements specification languages have not been chosen as the requirements which are dealt with have a more general character.

   While using natural language it will be focused on creating a clear and easily understandable requirement specification. Every potential reader should be able to understand the requirements specification without the need to familiarise with special specification techniques.

   Furthermore the centre of interest of this thesis is not only the creation of a requirements specification, but also the analysis of similar products which already offer a Web-based UI and the analysis of the Terminal client-server architecture as an alternative. Therefore not every requirement will be specified in detail, as it would go beyond the scope of thesis.

2.3.1. Unified Modelling Language

UML is an object modelling and specification language used in software engineering. It is officially defined by the Object Management Group. The following two views will be used for the creation of models during requirements analysis in section 3.

- Use Case View: "The use case view models the functionality of the system as perceived by outside users, called actors. A use case is a coherent unit of functionality expressed as a transaction among actors and the system. The purpose of the use case view is to list the actors and use cases and show which actors participate in each use case." [RJB99]

The use case view will be used in section 3.2 to illustrate the functions and the involved actors during the analysis of the current functions.

- Static view: "The static view captures object structure. An objected-oriented system unifies data structure and behavioural features into a single object structure. The static view includes all the traditional data structure concerns, as well as the organisation of the operations on the data." [RJB99]

  The static view will be utilised in section 3.2.8.1 to give a simplified model of the database structure of VP.

The tool Enterprise Architect 6.1 of Sparx Systems Pty Ltd. has been used to create the UML models. It was used as it is the product of choice at VP-GE for the creation of any kind of UML model.

2.3.2. 3LGM² Meta Model

The following subsections are a summary of several papers [WHBW04, WBW03, HBABW05] and book extracts [HAWB04], which should be consulted for further reading. If appropriate the descriptions are literal citations taken from the documents. Otherwise the actual text was shortened or paraphrased, still reproducing the ideas and opinions of the authors.

2.3.2.1. Introduction

A HIS is composed of heterogeneous paper- and computer-based subsystems and thus highly complex. Before the 3LGM^2 was created, no sufficient means to model a HIS as a whole and to show the dependencies between the subsystems existed. Especially for reasons of planning, directing and supervising, a blueprint or model of the HIS is indispensable [WBW03].

On account of these needs the Three-Layer Graph-Based Meta model (3LGM^2) was created by the IMISE to model HIS adequately. The Meta model is based on the UML [WBW03] and aims to support the systematic management of a HIS as well as the quality assessment of information processing in hospitals [HAWB04]. Apart from modelling, an additional intention while creating the 3LGM^2 Meta model was to define a precise terminology for designating the components to prevent communication problems.

The layers were created to distinguish between different levels of abstraction. The domain layer concentrates on the *enterprise functions* of a hospital, the logical tool layer emphasises on application components whereas the physical tool layer focuses on physical data processing components. The layers will later be described in more details.

The 3LGM^2 Meta model will be used for requirements analysis because it offers the best means to visualise the differences between the current state of integration of VP in the HIS and the future state. Modelling these two states with the 3LGM^2 Meta model will greatly contribute to the goal of creating a well-founded understanding of the problem domain.

Table 2.1.: 3LGM^2 meta model – graphical representation of elements at the domain layer.

Element	**Graphical representation**
enterprise function	rectangle
entity type	oval
using information	arrow from entity type to function
updating information	arrow from function to entity type

2.3.2.2. Layers and Their Relationship

Domain Layer: "The domain layer describes a hospital independent of its implementation as a set of enterprise functions. These functions need information of a certain type about physical or virtual entities of the hospital. These types of information are represented as *entity types*. The access of a function to an entity type can be *using* or *updating information*." [HAWB04, WBW03]
The graphical representations of the described elements can be found in 2.1.

Logical Tool Layer: "At the logical tool layer, *application components* are the centre of interest. application components support enterprise functions. Computer-based application components are controlled by *application programmes*, which are adapted *software products* (off-the-shelf products); paper-based application components are controlled by conventional working plans that describe how people use paper-based data processing components. application components are responsible for the storage and for the communication of data about entities of a certain type. Computer-based application components may have a local *enterprise function* to store data, and paper-based application components may file their documents in a *document collection*. Communication interfaces ensure the communication among application components (*component interfaces*), but also between an application component and a user (*UI*). For communication among application components, *communication links* can be defined." [HAWB04]
The graphical representations of the described elements can be found in 2.2.

Physical Tool Layer: "The physical tool layer is a set of physical data processing components. They can be human actors (such as the person delivering mail), paper-based physical tools (such as printed forms, telephones, books, paper-based patient record, administrative stickers), or computer systems (such as terminals, servers, personal computers, switches, routers). They are physically connected via so-called data transmission connections (for example data wires). The constellation of these connections leads to physical networks, which are based on network protocols. Note that physical as well as logical networks can be represented on the physical tool layer." [HAWB04]
The graphical representations of the described elements can be found in the documentation of the 3LGM^2 modelling tool. Physical data processing components can be illustrated by custom symbols to emphasise their type.

Inter-Layer-Relationships: "A variety of dependencies, called interlayer relationships, ex-

Table 2.2.: 3LGM^2 meta model – graphical representation of elements at the logical tool layer.

Element	**Graphical representation**
application component	rectangle with rounded corners
database system, document collection	cylinder
component interface, user interface	oval
communication link	arrow, pointing it the direction in which the data flows

ist among components of different layers. Relationships exist between classes of the domain layer and the logical tool layer and between classes of the logical tool layer and the physical tool layer." [HAWB04]

Application Component Configuration: Relationships between the domain layer and the logical tool layer are represented by the so-called *application component configuration*. It states which enterprise function is supported by which application component. It is possible that a function is supported by several application components, by a single application component or by combinations of both. Thereby, the application component configuration may give hints about redundancies (enterprise function is alternatively supported by several application components) or weaknesses (one or several application components are jointly necessary to support an enterprise function) in the domain layer.[HAWB04]

Data Processing Component Configuration: Relationships between the logical tool layer and physical tool layer are represented by the so-called *data processing component configuration*. It states which application component is installed on which physical data processing component. It is possible that an application component is installed on several (for example client-server based system), on one (stand-alone system) or in combinations of both physical data processing components.[HAWB04]

2.3.2.3. Reference Model for the Domain Layer of HIS

In [HBABW05] Hübner-Bloder et al. proposed a reference model for the domain layer of the HIS. It was created from the need to save a considerable amount of effort needed when modelling specific HIS. Moreover, it can be used as model patterns to assist with deriving more specific models from it and to directly compare models concerning their completeness.

The domain layer of the 3LGM^2 models which will be created in this thesis will be derived from the reference model of Hübner-Bloder et al. Until now only a German version of this model exists.

2.3.2.4. 3LGM² Modelling Tool

Apart from the 3LGM² Meta model, IMISE also develops a software tool called 3LGM² Tool to create and analyse 3LGM² models. It can be downloaded from `http://www.3lgm2.de`.

This tool will be used to create the 3LGM² models in this thesis. It still has beta status.

2.4. Methods of Information Procurement

Each method which has been used for information/knowledge procurement will be subsequently listed. In the beginning a general description of each methods will be given, followed by statements for which purposes they have been used. For details on the methods and their pros and cons please refer to [HA05].

2.4.1. Interview

"An interview is a conversation between two or more people where questions are asked to obtain information from the interviewee" [Wik06]. Interviews are appropriate to examine complex issues and offer a maximum of interactivity between the interviewer and interviewee, for example to eliminate ambiguities. A disadvantage is that this method is time consuming [HA05] .

This method has been used to gather information about single issues. For these issues it was already known that clear requirements exist. In addition, an interview was used in case that no adequate information sources have been found. As the diploma thesis is performed on behalf of VP-GE, the author had access to the knowledge of the employees at VP-GE. Furthermore, the thesis is performed in close cooperation with IMISE and the author also had access to the knowledge of the staff there.

2.4.2. Questionnaire

A questionnaire is a set of written questions which have normally to be answered in writing by more than one person. Possible answers can be standardised so that the persons completing the questionnaire can only choose from predefined answers. Thereby, a fast evaluation is possible and a maximum of comparability is achieved [HA05]. Disadvantages of standardised answers are that the respondents may not clearly understand the question or available answers and will therefore give a wrong statement on the issue. In case of free text answers, the respondents is given a maximum of freedom to answer the question adequately while significantly increasing the evaluation efforts. Without offering predefined answers, a maximum of information can be taken from the questionnaire, if answered thoroughly. Standardised answers may not consider certain aspects which are also relevant to the related issue.

The following two questionnaires have been used in the course of this thesis to assess information:

2.4.2.1. Questionnaire "Features of a Web-based Interface to VP"

The aim of this questionnaire was to find out, which functions should be offered by the Web client.

After a consultation with B. Hornung it was decided not to contact customers directly. This would have been connected with too much effort whereas most of the required information are already at hand of VP-GE. The following employees of VP-GE have been chosen for answering the questionnaire, as they all coincided with the demands for a Web client:

- *Wilson Gottschield (WG)*: He is head of the American sales team and therefore extremely close to the customers and their demands. He was chosen to represent the demands of the American customers.
- *Michael Nilles (MN)*: He is representative of the German sales team. He was chosen to represent the demands of the German customers.
- *Franz Stölzle (FS)*: He is the product manager of VP. Among other things his business is to collect and quantify feature requests for future versions and to develop the marketing strategy. He was chosen as he has a holistic view of VP itself and its further development.
- *Bernhard Hornung (BH)*: He is head of the development department. He was chosen as he has sound knowledge of the implementation efforts of most features and of how realistic it is to offer them through a Web client.
- *Konrad Pscheidl (KP)*: He is head of the customer service. He was chosen as he has distinctive experience of the usage of VP in a hospital in everyday life.

A questionnaire with standardised answers was chosen, as it was seen as crucial to assess the opinion of multiple respondents in an equivalent way and to create out of it an overview with an universally valid conclusion from it. The respondents could valuate each listed feature with integers between 0 and 3:

- *3* denotes highest significance, meaning that in case of an incremental stepwise approach with 3 version releases, it should be implemented in the first version of the Web client.
- *2* denotes medium significance, meaning that it should be implemented in the second version.
- *1* denotes minor significance, meaning that is should be implemented in the final version, offering all functions which the Web client is intended to have.
- *0* denotes that this features is not needed and should thus not be implemented by the Web client.

By calculating the integral rounded arithmetic means of the values given by the different respondents to one feature, the over-all importance of the related feature was assessed.

One part of the questionnaire lists features offered by the Native client. These features are a summary of all features specified by the requirements specification [Seh05] and described

in section 3.2. Certain features have been pooled into one single feature. This was done since it would take too much time to judge the importance of each feature and also because many features are interconnected, meaning that the implementation of one feature would go along with the implementation of another one. The results of this part have been used in section 5.2.1.

The other part of the questionnaire lists additional features, which could be offered by the Web client. The results of the second part have been used to answer certain issues in section 5.2.

The questionnaire in its original form together with the individual answers of the respondents can be found in the appendix.

2.4.2.2. Questionnaire "Product Analysis"

The aim of this questionnaire was to find out how the requirements which will be set up in this thesis are realised by several products. Details which product has been chosen and reasons why can be found in section 2.5.

The questionnaire has been sent by email to the following persons in agreement with B. Hornung:

- *John Stepleton* of Euroking Miracle Ltd.: He is the development director at Euroking Miracle Ltd.. He was chosen as he has a sound knowledge of E3 and its implementation. The questionnaire has been answered in the course of a phone interview with J. Stepleton.
- *Svein Fimreite* of GE Vingmed Ultrasound: He is the developer of WebEx and part of the development team of EchoPAC. He was chosen as he has a sound knowledge of it and its implementation. The questionnaire has been answered in writing by S. Fimreite. Ambiguities in the answers have been clarified via email.
- *Christian Harböck* of Trium Analysis Online GmbH: He is developer of Trium CTG Online. He was chosen as he has a sound knowledge of it and its implementation. The questionnaire has been answered in writing by C. Harböck. Ambiguities in the answers have been clarified via phone and email.

A questionnaire with free text answers was chosen, as it was seen as crucial to take as much information from the answers as possible. At the time the questionnaire was created, the requirements had not yet been fully developed. So it was important to allow free text answers because thereby aspects yet unknown could also be considered. Nevertheless, it turned out that all important issues had already been covered by the requirement specification draft.

The questionnaires in its original form together with the individual answers of the respondents can be found in the appendix.

2.4.3. Analysis of Existing Data

The analysis of existing data was extensively used throughout this thesis. Apart from literature research in which books and papers were consulted, also manuals and other technical

documentations at VP-GE were accessed. For additional and more up-to-date information the Internet was consulted. The related resource are specified in the bibliography of this thesis.

2.5. Product Analysis

In case a new application component is needed in a business, a requirement specification is set up. According to this specification products at the market are analysed in respect to their compliance. Market analysis should be performed by utilising a data entry form which could be a brief summary of the requirement specification. By using a data entry form a maximum of comparability is achieved during product analysis. The results may help to refine or revise the original requirements specification. [HA05]

In this thesis only an analysis of certain predefined products was carried out. The products to be analysed have been defined by VP-GE. Namely they are:

- E3 of Euroking Miracle Ltd.,
- WebEx of GE Vingmed Ultrasound and
- Trium CTG Online of Trium Analysis Online GmbH.

Selection criteria have been that the product characteristics are similar to VP and that the manufacturers would cooperate in providing the required information. Product analysis will not be performed with the intention to pick a certain product. It will be done to obtain a detailed idea of how other companies coped with certain issues of the requirement specification and where they saw opportunities or problems in offering a Web client.

Information has been gathered in form of a questionnaire, see section 2.4.2.2 for details.

The results will be given in section 6.2.

2.6. Native Client versus Web Client

In oder to highlight the differences between a Native client and a Web client, a short characterisation of each client architecture will be given.

In this context the term client designates a software component (programme) with a separate UI used to access and display data which is managed and stored by a central server. The clients run on separate workstations and communicate with the server over the network. This kind of architecture is known as the client-server architecture. Clients and server together form a system which is designated as software product.

Data which is exchanged between client and server will be distinguished into user data and control data for an easier understanding. User data designates the actual data which is relevant and meaningful to the user (for example patient information). Control data designates data which is used for providing the UI (for example layout definition of the UI, underlying business logic).

Albrecht et al. proposed in [Alb03] the following criteria for comparing different client architectures:

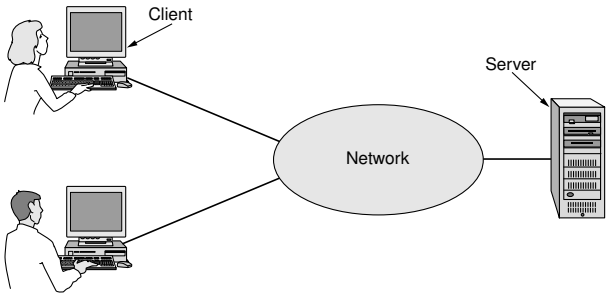


Figure 2.3.: Client-server architecture, taken from [Tan03].

- *Cross Platform:* Aims to examine to which extend the client architecture supports different Platforms.
- *Distributed:* Aims to examine to which extend the client architecture supports distributed usage.
- *Interactive:* Aims to examine the degree of interactivity between the user and the system.
- *Rich Interface Components:* Aims to examine the functional range of the UI which could be provided by the related client.

After giving a general description, these requirements will be used to achieve a good level of comparability while characterising the Native and Web client.

2.6.1. Native Client

A Native client is a UI provided by a dedicated programme. The programme is compiled for a certain Platform. It can utilise all OS functions and so also control the local hardware of the workstation, for example serial interfaces or 3D hardware acceleration.

In most cases the UI's layout and the underlying business logic is defined and processed at the client side. All or at least a set of files needed for the execution of the Native client (control data) is transfered from the server every time the client is started. When the client is running mainly user data is transfered between the client and the server.

- **Cross Platform**
  - Clients are typically targeted at single Platforms and therefore easier to use and faster to implement. [Alb03]
  - Most clients are restricted to a certain Platform, but can be ported to other Platforms with the utilisation of cross-Platform libraries (like Qt). [Alb03]
- **Distributed**
  - Native clients are characteristic for traditional group-ware systems. Those systems are designed for Local Area Network (LAN) usage and thus have high-bandwidth utilisation . [Alb03]

  - Clients require to be installed on each workstation. Complex settings for each workstation have to be configured properly. [Alb03]
  - If a new version of the software product is issued, the clients have to be reinstalled or updated.
  - Once installed, in most cases user data is the only type of data which is exchanged between the client and the server.

- **Interactive**
  - Clients are most responsive as the logic responsible for the UI is controlled and executed on the client side.
  - Only parts of the UI can be updated quickly based upon user interactions.
  - In general application, screens are dynamical and predictable. [Alb03]

- **Rich Interface Components**
  - Clients offer access to all components of the Platform as they are native. [Alb03]
  - Users feel at home as the client's interface works like those of other applications on their OSs. [Alb03]

2.6.2. Web Client

A Web client is a UI which is Web-based and displayed by a Web browser. Web browsers are available for all platforms. They utilise only few OS functions (like accessing the local file system) and offer no means to control the local hardware of a workstation.

In most cases the UI's layout and underlying business logic of the UI is defined, created and processed on the server side and made available in form of a so called Web page to the client. During startup and runtime of the Web client, Web pages containing user and control data are exchanged between the client and server.

Originally only static Web pages have been stored on the server (called Web server). With the utilisation of scripting languages, Web pages are created dynamically by the server at the moment they are requested from the client. The main advantages of this procedure are an easier maintainability of the server and better customisation to the user needs. The Web browser fetches the Web page and displays it to the user. Interactivity is achieved by the usage of links. Each click of the user on a linked object inside the Web page causes the Web browser to send a request with the related URL to the server. The server normally answers this request by sending an updated or new Web page to the Web browser, depending on the URL clicked. Due to these constraints it is clear that the performance of this client architectures relys strongly on the quality of the network connection.

The core part of a Web browser is the layout engine. It creates a well formatted UI from a Web page which contains user data and markup information (like `Hypertext Markup Language (HTML)`) on the basis of formatting information (like style-sheets). The recognition and interpretation of markup is dependent on the Web browser. Even if the Web page conforms to certain standards, it may occur that, Firefox for example is able to display the page and Microsoft Internet Explorer cannot display it.

In order to be able to display and process also more advanced content, certain techniques have been created to extend the limited functionality of a Web browser. These extensions

can be divided into the following two types: Plug-in and Helper Application [Tan03]. Both have in common that they are compiled for certain platforms and therefore are platform specific.

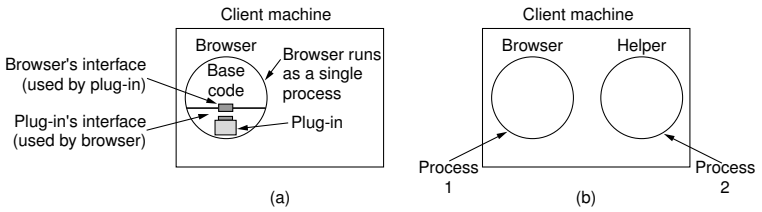


Figure 2.4.: Web browser plug-In and helper application, taken from [Tan03].

Plug-ins are code modules which are executed within the Web browser's process, they have access to the current Web page and can thus modify its appearance. The interface between Web browser and Plug-in is Web browser specific. The Plug-in has to be installed so that the Web browser is able to call it. [Tan03] Examples for Plug-ins would be Java Script or Macromedia Flash.

Helper Applications are complete programmes, running as separate processes. They have no interface with the Web browser and usually open a certain file specified by an URL. The Helper Application has to be installed [Tan03]. Examples for Helper Applications would be Adobe Acrobat Reader or Java.

- **Cross Platform**
  - Web pages can be displayed by a Web browser on all platforms. [Alb03]
  - Browser incompatibilities cause problems.
  - Usage of Web browser extensions limit the cross platform support. Plug-ins and Helper Applications are platform dependent, Plug-ins are additionally Web browser dependent.
  - Require target platform testing and related optimisation. [Alb03]
- **Distributed**
  - The Client architecture was originally designed for a distributed environment.
  - Clients do not have to be installed, as Web browsers are available on all Platforms by default.
  - `HTML` protocol is thin and transfers quickly over low bandwidth connections. This point does not apply on highly-interactive applications, which require frequent refreshing of the page to achieve interactivity and potentially cause significant server load and redundant data transmission [Alb03].
  - User data and control data (markup and formatting information) is exchanged between the server and the client.
- **Interactive**
  - Web pages are essentially static once they leave the server. Screen updates are not shown immediately and require a periodic client-initiated refresh. [Alb03]

  - Web browser extensions offer means to improve interactivity.

- **Rich Interface Components**
  - UI components are severely limited. [Alb03]
  - Web browser extensions are required to offer full-featured UI.

2.6.3. Comparison

In table 2.3 a brief comparison of both client architectures is given, stating which type of client has poor ("-") or extensive ("+") support of the related criterion.

Table 2.3.: Comparison of Native client and Web client architectures.

Architecture	Native client	Web-based client
Low bandwidth utilisation	-	+
Control hardware	+	-
Cross platform	-	+
Distributed	-	+
Interactive	+	-
Rich interface components	+	-

2.7. Terminal Server

The so called Terminal server technology is a third type of client-server architecture which is articulately different to the two architectures described before.

In order to recall their main characteristics: In case of a Native client, all of the computation needed for creating the UI and realizing the business logic behind it is performed on the client side. During startup, mostly control data is transfered between the client and server, during runtime mostly user data is exchanged. In case of the Web client, most of the computation needed for creating the UI and realizing the business logic is done on the server side. During startup and runtime control and user data are exchanged. Still, certain computation is performed at the client side to display the Web page through the Web browser, possibly by the utilisation of certain Web browser extensions.

A Terminal client works quite different to these approaches. In a Terminal client-server architecture, multiple users are given the possibility to remotely work by using Terminal clients on a certain computer, the Terminal server. Each of the connected users is provided with her/his own UI, which is giving them the experience of sitting in front of the server and interacting directly with it. This is due to the separation of input (keyboard, mouse) and output devices (screen) from the actual computer. A Terminal client retrieves and displays the screen from the Terminal server and forwards the related input actions (key strokes, mouse movements) to it. All computation (like programme execution) is performed on the Terminal server. Each connected client is given an equal part of the server resources to do so. As the server resources are partitioned to the connected users, multiple users can work on the same server without interfering with each other. By using the Terminal

client the users can access the same functions offered by a workstation, meaning that they can interact with the OS and execute any application component which is installed on the Terminal server, for example the Native client of VP.

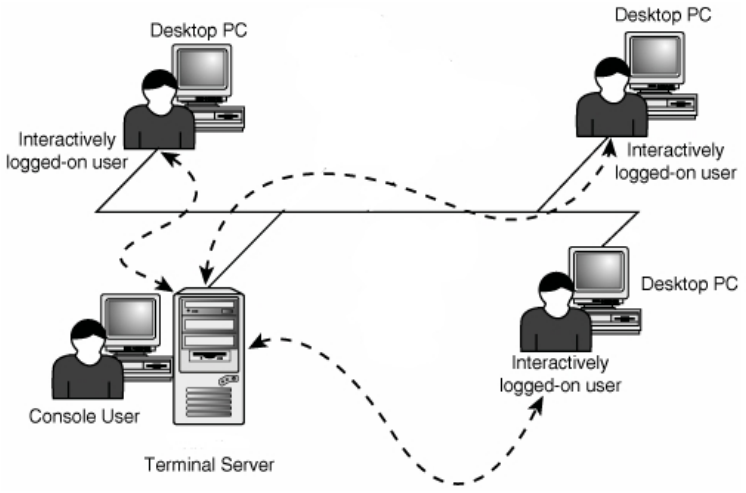


Figure 2.5.: Terminal server with one local and three remotely connected users, modified version of a figure taken from [Mat04].

In figure 2.5 a Terminal server to which three users are remotely connected via Terminal clients is shown. The clients are running on the three workstations which are denoted as Desktop PCs. One user is directly accessing the Terminal server via its UI. The UI is denoted as Console.

In order to continue the characterisation of the different client types as started before: In case of the Terminal client, all of the computation needed for creating the UI and realising the business logic is performed on the server side. The only data transfered during startup and runtime is control data.

For forwarding the input and output actions only a minimum of hardware equipment is needed. On the one hand a Terminal client could only be software. Here the Terminal client is a standard programme executed on a workstation, which has its own OS and other application components running. On the other hand, a Terminal client could be hardware and software with the only purpose to act as Terminal client.

In contrast, the Terminal server has high hardware requirements, as it has to support multiple users at once. Each of the connected user may execute different programmes at once. So a fast Central Processing Unit (CPU) and much of RAM is required.

During the old times of mainframe computers, the UI provided to the Terminal client was just a console (text-based, non-graphical UI). Nowadays Terminal servers offer a fully functional graphical UI to their clients. Representatives of these systems are the Microsoft Windows Server 2003 Terminal Services or the X Windows System system of Linux. These Terminal server solutions do not only support the forwarding of standard input/output devices like screen, mouse and keyboard, but also enable the forwarding of advanced interfaces like serial ports or sound output.

2.8. Clinical Context Management

Most of the information which will be given in this section are taken from [Sel01] and [Sen00]. For more information on CCOW please refer to HL7 Australia CCOW Resources Page at `http://www.hl7.org.au/CCOW.htm`.

CCOW can be described as a standard aimed at offering proper means for contextual integration at the point of use. It is issued by the Clinical Context Object Workgroup of the HL7 Committee, which is already well-known for the HL7 standard for clinical communication. With the utilisation of CCOW the user has the experience of interacting with a single system, when in fact he or she may be using multiple, independent application components each via its native UI.

By synchronising and coordinating application components so that they automatically follow the user's context, for example displaying data of same patient in every application component running on the workstation, CCOW serves as the basis for ensuring secure and consistent access to patient information from heterogeneous sources. Generally speaking CCOW offers means to synchronise user and patient data from different application components. By synchronising user data CCOW also offers means for Single Sign-On (SSO) [1] solutions.

Benefits of CCOW are that

- it is easier to use,
- it offers increased utilisation of electronical information,
- it facilitates an increase in patient safety,
- and offers means for secure context management (like it has been addressed by Health Insurance Portability and Accountability Act (HIPAA) requirements).

CCOW's Context Management Architecture (CMA) is composed of the following components:

- application components (for example VP or SAP IS-H*med).
- Context manager which coordinates and synchronises application components. It is founded on the principle that common context can be established across applications by identifying things (for example a patient) [2] or concepts (for example a clinical encounter) [3] in a manner that different applications can nevertheless recognise them.
- Mapping agents which represent various synonymous real-world identifier used to identify clinical patients, users or other subjects[4]. As application components may

[1] SSO denotes a technique with which the user has to log on just once to be authorised for using different application components running on the workstation.

[2] In the context of data modelling a thing (for example a patient) would be an entity type.

[3] In the context of data modelling a concept (for example a clinical encounter) would be an attribute type of an entity type.

[4] In the context of CCOW objects on whose basis the context is synchronised from different application components are denoted as subject. A variety of subject types are available, for example user identity, patient identity, encounter identity, observation request identity, DICOM study identity etc.

use different identifiers (for example the hospital's patient ID versus an application component specific ID), the mapping agents offer means to map between the different context subject identifiers, so that an application component only needs to know the identifiers of its own to establish the context.

- Annotation agents which supply ancillary data about an identity subject. Ancillary data could be for example patient demographics, if the identity subject is the patient. Each hospital defines the data sources for its annotation agents and so assures data integrity. They have been added in CCOW version 1.3.

- Action agents are proposed to be added to the CMA. They should be enabled to provide non-repudiation of document content and irrefutable evidentiary status of an individual's identity and professional credentials to enable digital signatures. Yet, this is only a proposal specified by [Wil03].

It is vital to note that the CCOW CMA does define the roles and responsibilities for each of the listed components. Moreover, it precisely describes the interfaces that enable them to communicate. It does not, though, define the implementation of any component.

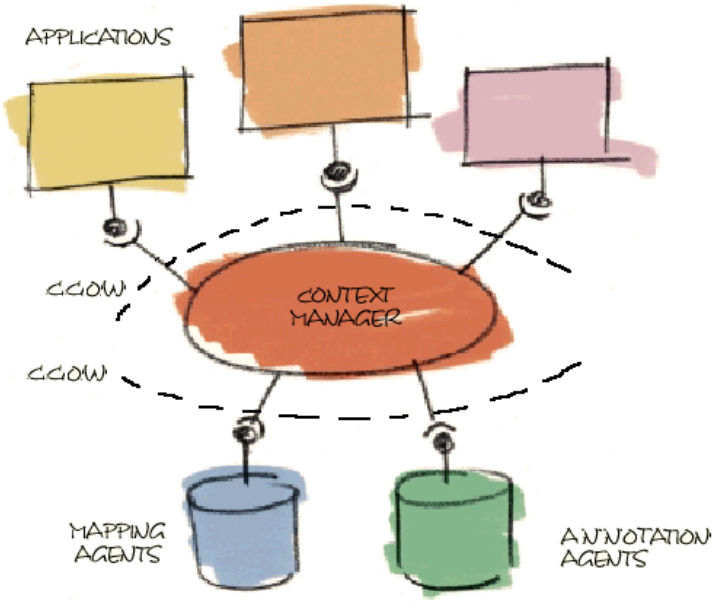


Figure 2.6.: Overview of the components of the CCOW CMA, taken from [Sel01].

In order to illustrate how CCOW works in detail, all steps which are required to synchronise the context between different application components are subsequently listed in their order of execution. This listing is a literal citation taken from [Sen00]:

1. User selects the patient of interest from any application on the clinical desktop.

2. Application tells the context manager to start a context change transaction and sets the context data to indicate the newly selected patient.

3. Context manager tells patient mapping agent that context change is occurring; mapping agent supplies the context manager with other identifiers by which the patient is known.

4. Context manager tells demographics annotation agent that context change is occurring; annotation agent supplies context manager with authentic demographics data.

5. Context manager tells the other applications that a new patient context has been proposed. The context manager surveys the applications to determine whether each can apply the new context.

6. Each application indicates whether or not it can apply the new context.

7. If one or more of the applications cannot or prefers not to apply the new context, the user is asked to decided whether to continue, or cancel.

8. Context manager tells each application to apply new context, or that the transaction has been cancelled.

9. Each application applies the new context if instructed to do so by the context manager. Each application gets the new patient context from the context manager.

CCOW is still a young standard and has been severally revised since its original release. With each revision new functions have been added. Following, only additions relevant for context integration of Web-based application components will be listed:

- Since version 1.2 CCOW also supports Web-based application components. Web-based application components and mapping agents can for example send and receive URL-encoded `HyperText Transfer Protocol (HTTP)` messages to the context manager. This version also comes with the capability to deploy over private and public networks, for example the utilisation of the `Secure Socket Layer (SSL)` for communication over public networks.

- In version 1.5 CCOW is proposed to support `Simple Object Access Protocol (SOAP)`. In general, `SOAP` can be used for exchanging `Extensible Markup Language (XML)`-based messages over a computer network, usually by using `HTTP`.

According to the author's knowledge, CCOW version 1.5 still has draft status and CCOW version 1.4 is the last official version.

As stated above, CCOW is still an emerging standard. In spite of extensive literature research, no useful reports have been found which aimed at analysing or evaluating the practicability of CCOW.

3. Requirements Analysis

3.1. Outline

In the preceding chapter the basic principles have been described.

The aim of this chapter is to provide a detailed understanding and knowledge of the problem domain. First of all, the current functions of VP will be described in section 3.2. Then, the current integration state of VP within the HIS will be analysed in section 3.3. Therefore, the interfaces offered by VP will be identified and the current state of integration will subsequently be modelled. This chapter will end with identifying weak points of the current solution (the Native client of VP) and weak points which are connected to the future solution (the Web client of VP) in section 3.4.

The succeeding chapter will define out of the discovered weak points a requirement definition for the future Web client.

3.2. Current Functionality of the Native ViewPoint Client

This section is meant to give the reader a well-founded idea of the functions offered by the Native client of VP. The purpose is not to give a complete and detailed requirements specification or system description of the Native client and the VP system in general.

In order to give an example: At this point it is of importance, that the client offers a DICOM interface, but it is negligible how the import is performed in detail (for example what happens if the received DICOM object already exists in VP).

For a detailed description of each feature, its usage and its related behaviour, please refer to the original requirements specification [Seh05] or the user manual [Rie05b].

In order to gain a more general idea of what VP is, please refer to chapter 1. For a detailed description of how VP is integrated in the HIS, please refer to section 3.3.

The analysed version is VP 5, which is the latest version.

3.2.1. Hardware and Software Requirements of a Client

Before starting with the actual description of each module, the basic requirements of a VP client will be mentioned, as already addressed in the introduction.

On the software side, the VP client and server are designed to run only on Microsoft Windows versions 2000, XP and 2003 Server. Not all Service Packs of the related Microsoft Windows versions are supported. In order to display videos VP needs at least Microsoft DirectX 8.1. The only supported Database Management System (DBMS) is Sybase SQL

version 9.0.

On the hardware side, the workstation on which VP should be installed, needs to fulfil a set of minimum hardware requirements. For example, when the client is intended to record video sequences at least a computer with a 2.5 Ghz CPU, 512 MB RAM, 40 GB Hard Disk Drive (HDD), 100 MBit/s `TCP/IP` network connection, free serial ports and a full length Bus Master PCI Slot is required. It should be noted that in general a 100 MBit/s `TCP/IP` network connection should be preferred. However clients can already operate on a 2 MBit/s line, resulting in a loss of performance. These relatively high bandwidth requirements are due to the fact that VP was originally designed for application in LANs only.

For detail information on the requirements of VP version 5, please refer to the system requirements sheet [Rie05a].

3.2.2. Installation Procedure

In general, if VP should be installed on a workstation, the following steps have to be executed before the Native client is ready for usage: First, it should be checked if the client meets the criteria of the minimum software and hardware requirements. Then the VP software client needs to be installed and configured properly. During this step a valid license (string) has to be assigned. If the client's functions were tested successfully, VP is ready for productive usage on this workstation.

If an update for VP is issued, it has to be installed on the VP server. The next time the VP clients are started, they automatically retrieve all updated files from the server. High bandwidth is required for proper updating of the client, as the size of the transfered files is usually 50 MBytes or more.

3.2.3. General Description

VP offers functions of a MDS and of a PACS and can be seen as a combination of both. It is a PACS as it offers extensive functions for image data editing, communication and archiving. For more details please refer to 3.2.8.3. Concerning the types of MDSs, it can be clearly assigned to the following classes defined in [LGH+03]:

- *Class I1*, documentation with primary clinical information (factual knowledge);
- *Class P1*, documentation for a primary patient related reporting;
- *Class S3*, mostly standardised documentation;
- *Class U1*, primary vertical documentation (few data objects with many attributes);
- *Class D2*, primary direct documentation and
- *Class R1*, computer based documentation.

While offering functions of two application component types, it is crucial to note that in case of VP the diagnose is created from the acquired image data and, thus, only findings

and image data in combination have medical significance. So for the user it is most preferable to have access to both data types (findings and image data) at once and through one application component.

In order to now describe the actual functions of a VP client, the different modules of VP will be analysed. Their availability to the user is controlled by licensing (client specific license defines which modules are available to the client) or by rights guaranteed to the currently logged-in user (user specific rights define which modules and functions are available to the user; VP has its separate user management). All listed modules are accessible via the main programme and will open in the same window, except for the system administration module (VPAdmin) which is a separate programme.

As the functions and usage of each modules and submodules are closely related, a 100 percent clear distinction cannot be made. The partitioning given here is mostly based upon the original requirements specification of VP version 5.0 [Seh05] and the former requirements specification of VP version 3.31 [Psc02]. Special emphasis was placed on making a novice acquainted with the different features offered by each module and with how the modules and features are interrelated.

3.2.4. Patient Management Module

VP from its purpose is a vertical documentation [LGH+03] and stores data in a patient oriented way. Therefore, a module is needed to manage and store medical findings and all related information patient specifically. The Patient Management Module fulfils these requirements and also supports the management of patient demographics.

In order to support the work flow of a functional diagnostic unit or doctor's practice in an appropriate way, patient management is offered by a separate module. This was done because there may be dedicated personnel to record and manage patient demographics. Since in a hospital several patients may have the same assigning physician, a list of all assigning physicians is maintained patient overlapping. One or more entries can be connected to a certain patient record.

Use cases of the management of patient records are shown in figure 3.1.

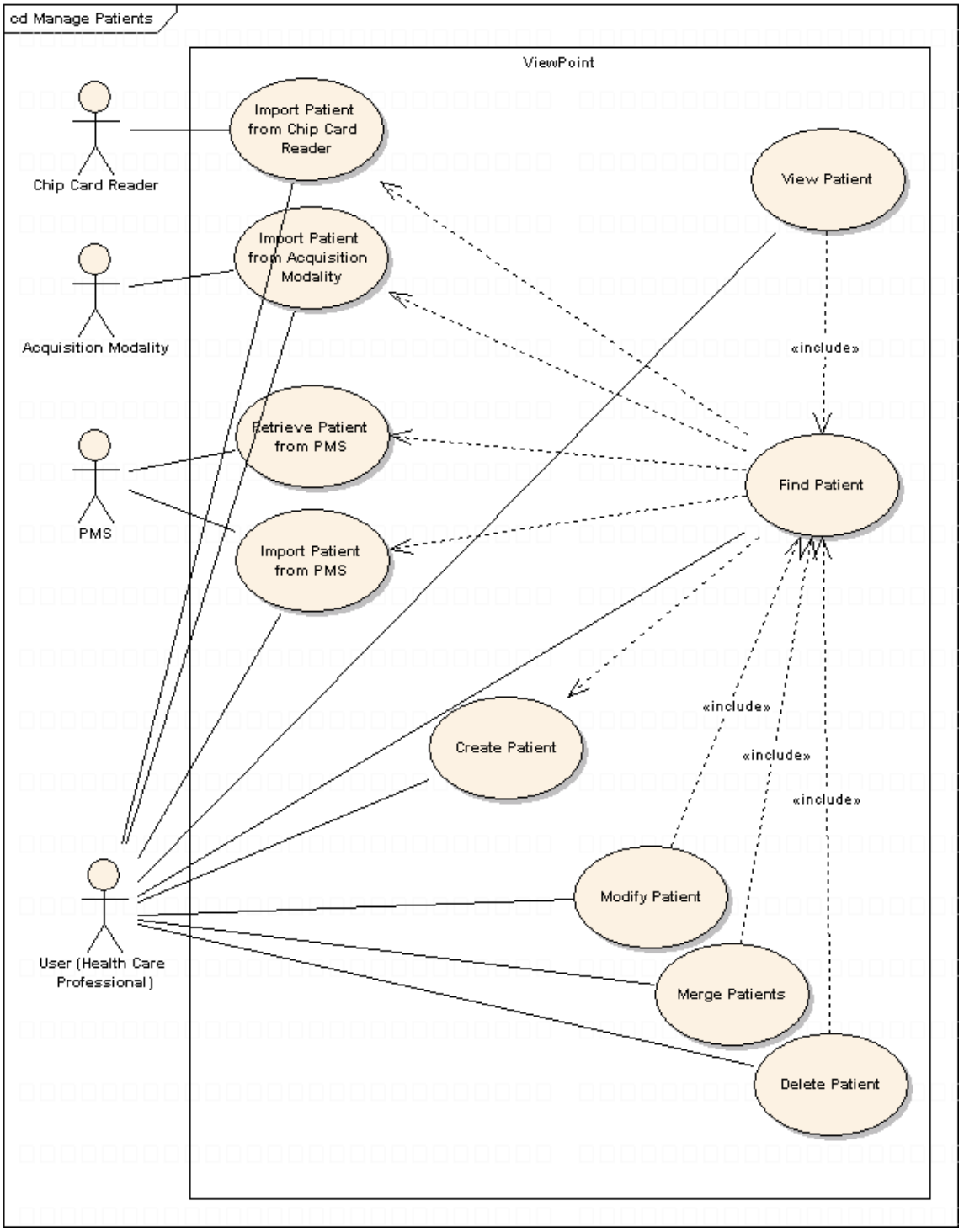


Figure 3.1.: Patient Management Module – UML use case model of the management of patient records, modified version of the model taken from [Seh05].

Offered functions for the management of patient records are:

- Create a new patient record.
- Find and view a patient record.
- Find and modify a patient record.
- Find and delete a patient record.
- Find and merge a patient record.
- Find and comment on a patient record.
- VP can import patient demographics from other application components (like PMS), if the corresponding interface is set up (see section 3.3.1.1 for details) to find or create a patient record. Variants are:
  - User initiated import, by pressing import button.
  - Automatic retrieval, like reception of a clinical order (orders are sent to VP and user can choose from a list which order to process).
- In case a DICOM interface (see section 3.3.1.2 for details) is used, VP can receive studies and find or create a patient record from the included patient demographics.
- In case a chip card reader interface (see section 3.3.1.7 for details) is used, VP can read patient demographics from an health insurance card and find or create a patient record from it.

Offered functions for the management of the list of the assigning physicians are:

- Create assigning physician entry.
- Modify assigning physician entry.
- Merge assigning physician entry.
- Deletion of assigning physician entry.
- Create or modify assigning physician entry from patient demographics included in a clinical order.

3.2.5. Patient Scheduling Module

VP has an integrated customisable schedule for planning examinations dependent on the availability of physicians and resources.

The schedule is used to manage appointments for examinations. An appointment can be made for a resource and a physician and be linked with a patient record. Resources can be rooms or medical devices.

Use cases of the Patient Scheduling Module are shown in figure 3.2. Offered functions are:

- Find appointment.

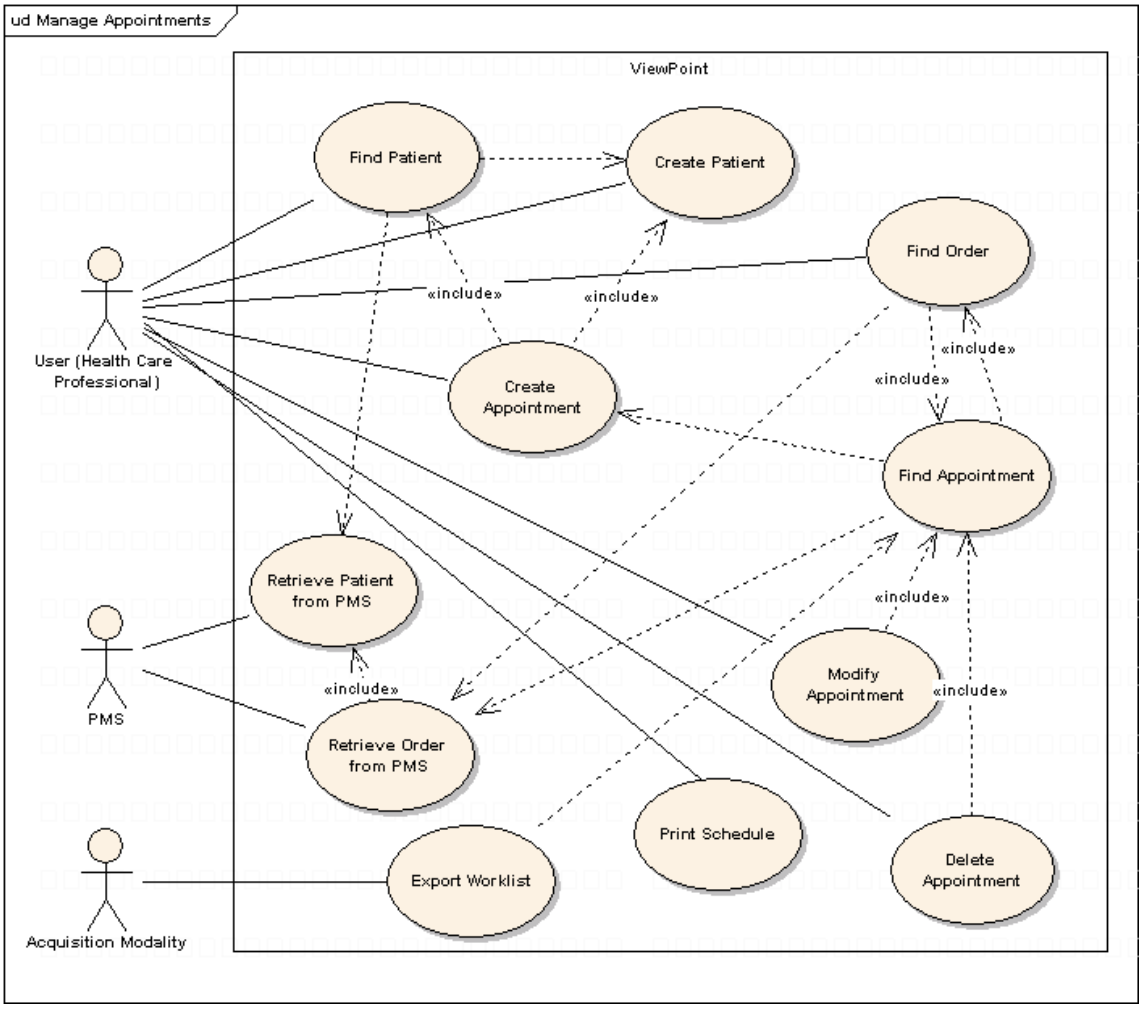


Figure 3.2.: Patient Scheduling Module – UML use case model of the management of appointments, modified version of the model taken from [Seh05])

- Create appointment. Appointments can also be created from orders.
- Modify appointment.
- Delete appointment.
- Print schedule.

3.2.6. Statistics Module

The statistics module is meant to offer the health care professional a way to vertically [LGH+03] analyse the recorded data. It offers a patient-overlapping view of the recorded findings and is, therefore, able to answer questions concerning clinical issues adequately. As medical data is entered in structured masks, which consist of data fields stored in a relational database, nearly every attribute of a patient can be accessed via this module by executing queries on this database.

The statistics module does not offer any professional statistic analysis functions as the name might suggest. It only offers functions to gather, densify and export recorded data, so that it can be statistically evaluated by an external application component like SPSS, for example.

Offered functions are:

- Menu-based definition and editing of new or existing queries.
- Predefined statistic queries for special purposes, for example standard statistics of certain health care regions or for certain institutions (for example Fetal Medicine Foundation London (FMF) London, see section 3.3.1.10 for details).
- Definition and modification of queries on a database language basis.
- Copy and export/import query definitions to/from the file system.
- Specification of filters depending on attribute types.
- Execute queries and display their results, print them out or export them as a file. ASCII, dBASE and Microsoft Excel formats are available.
- Search image data depending on its comments or search terms.

3.2.7. Archive Module

The purpose of the archive module is to offer read-only access to patient records and to enable tracing modifications made in them in the past.

Offered functions are:

- Every function from the examination module (see section 3.2.8 for details) which does not modify any data.
- History view which displays a chronological list of all modification done to a certain patient record.
- Ability to view which user made these modifications and when they have been made.

3.2.8. Examination Module

The examination module is the core of VP. Every function provided by the submodules listed here can be accessed through it.

3.2.8.1. Case and Examination Management Submodule

The case and examination management submodule is meant to offer the user the ability to manage the cases and related examination data of a patient. A case can be, for example, a pregnancy. Examinations are in this case all examinations and other procedures performed and documented within the scope of a pregnancy. So logically multiple examinations are

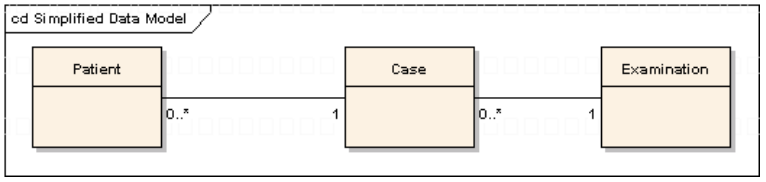


Figure 3.3.: Simplified UML data model of ViewPoint

related to one case, multiple cases are related to one patient. A simplified data model of VP is shown in figure 3.3, representing this relationship.

Furthermore, the case and examination management submodule offers functions to lock examinations, so that it is impossible to modify locked examination records. This is crucial to guarantee data integrity when examination data of patient records has been exported. For example this feature is essential for the German perinatal inquiry or other third party application components, where no way to instantly update the exported data exists if modifications in the patient record are made after the export process.

Offered functions for case management are:

- Create new case record.
- Find and view case record.
- Find and modify case record.
- Find and lock/unlock case record.
- Find and delete case record.
- Export case data to the cardiotocograph monitoring system Trium CTG Online.
- Process patient demographics as described in 3.2.4.

Offered functions for examination management are:

- Create a new examination record.
- Specify the type of the examination. Depending on the chosen type, one of the findings masks listed under 3.2.8.2 is being displayed.
- Find and view an examination record.
- Find and modify an examination record.
- Find and merge an examinations record.
- Find and lock/unlock an examination record.
- Find an examination record and assign/unassign a study (and its image data) received via DICOM to it (see section 3.3.1.2 for details).
- Find an examination record and assign/unassign a clinical order received from another application component to it (see section 3.3.1.1 for details).
- Find and delete examination record.

3.2.8.2. Findings System

The Findings System offers all functions needed for recording, displaying and processing medical findings in VP.

As the findings system is quite complex, a simple listing of the available functions is not seen as sufficient. So first a general description of the architecture will be given, followed by information about the report generated by the findings system. Afterwards the archive process will be characterised and the available findings masks for different areas of medical reporting will be listed. At the end, a list of the available functions will be given, differentiated by basic and extended functions. Basic functions are part of the findings system. Extended functions are connected to the Findings system, but play a more supporting role in it. In order to guarantee that the list is complete, functions already mentioned in the architectural description will be repeated.

Architecture: The findings are entered in structured masks. Their structure and content is defined by guidelines published by respective professional societies or created in close cooperation with health care professionals. In order to enable highest clarity, more detailed data on specific medical objects is represented in submasks which are accessible through the main masks (for example general findings on the heart would be entered in the main mask, more specific information about the heart valves would be entered in a submask). The content of the masks and submasks is composed of fields for different purposes (for example label, input field, link to submask etc.). Input fields are of different type (for example date, numeric value, text etc.) and can contain predefined values (for example predefined values of the examination status of the heart: "normal", "conspicuous", "cannot be examined", "not examined") as lists, from which the user can choose one or more values when completing masks. Depending on the entered input values, other fields can be automatically hidden or be displayed during runtime (for example if the examination status of the heart is "not examined", the link to the submask to enter details on the heart will be hidden) to achieve more clarity.

In order to support the user in estimating the meaning of numerical values, the findings module also offers functions to display graphs of the statistical distribution of most values. The graph data originates from scientific publications.

A sample screenshot of a submask is shown in figure 3.4.

The layout of masks, submasks, fields, graphs and their behaviour and coherence is defined in configuration files, which are interpreted during runtime.

ViewPoint Report: A report is automatically generated from the content of the masks, containing major information about the related examination. Its generation is also defined in the configuration files. It can be manually modified to the user's needs. Following, it will be designated as ViewPoint Report (VP-Report).

VP-Report is utilised by the Printing and Faxing Submodule (see section 3.2.8.4 for details) or when exporting medical data to other application components (see section 3.3.1.1 for details) like the CDMS. Furthermore, it can be saved as a file. A sample screenshot of the VP-Report is displayed in figure 3.6.

Changes made in a VP-Report can be traced in the Archive Module (see section 3.2.7 for

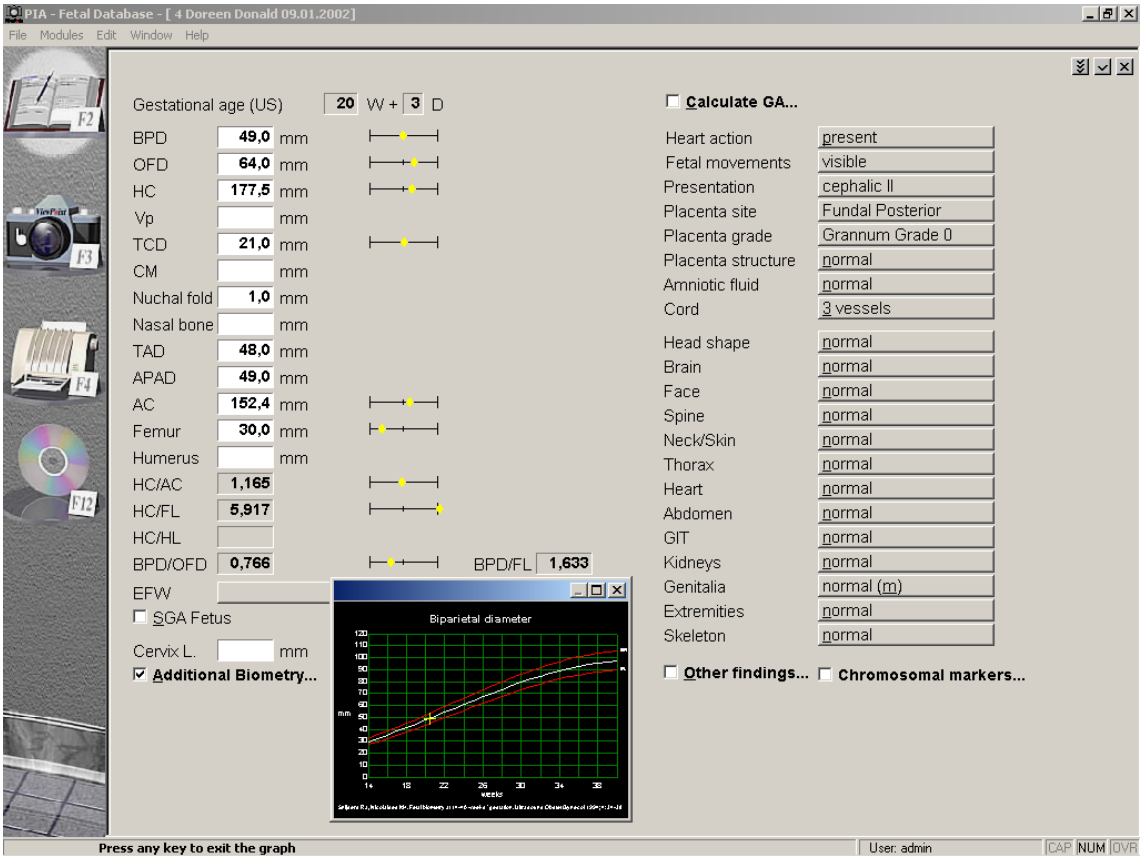


Figure 3.4.: Findings System, Prenatal Reporting – screenshot of the biometry submask.

details).

Use cases of the report handling are shown in figure 3.6.

Archiving Process: The user can archive the currently recorded examination (findings and image data) by pressing the "archive button" in VP. Although this step looks easy from the user level, several complex functions are executed at the system level. Some of these functions should be highlighted here.

Use cases of the archiving process are shown in figure 3.7. Executed functions are:

- Offer user the possibility to lock examinations after archiving.
- Store all data of findings and references to image data in the local database.
- Archive image data to local archive or if connected send it via DICOM interface (see section 3.3.1.2 for details) to a PACS. As archiving paths local directories as well as network shares which are handled like local archives are supported.
- Export billing data of the current examination like coded diagnoses and procedures to PMS.

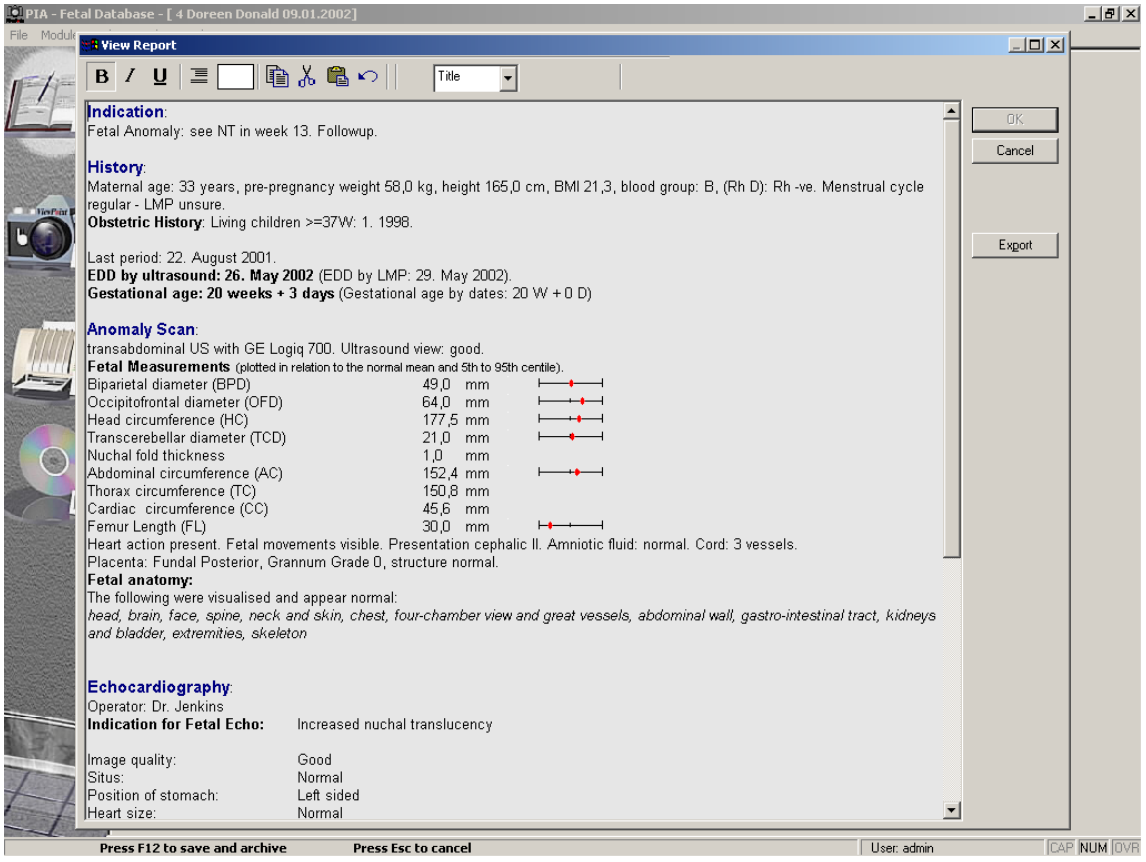

PIA - Fetal Database - [4 Doreen Donald 09.01.2002]
View Report
Title
OK
Cancel
Export
Indication:
Fetal Anomaly: see NT in week 13. Followup.
History:
Maternal age: 33 years, pre-pregnancy weight 58,0 kg, height 165,0 cm, BMI 21,3, blood group: B, (Rh D): Rh -ve. Menstrual cycle regular - LMP unsure.
Obstetric History: Living children >=37W: 1. 1998.
Last period: 22. August 2001.
EDD by ultrasound: 26. May 2002 (EDD by LMP: 29. May 2002).
Gestational age: 20 weeks + 3 days (Gestational age by dates: 20 W + 0 D)
Anomaly Scan:
transabdominal US with GE Logiq 700. Ultrasound view: good.
Fetal Measurements (plotted in relation to the normal mean and 5th to 95th centile).
Biparietal diameter (BPD) 49,0 mm
Occipitofrontal diameter (OFD) 64,0 mm
Head circumference (HC) 177,5 mm
Transcerebellar diameter (TCD) 21,0 mm
Nuchal fold thickness 1,0 mm
Abdominal circumference (AC) 152,4 mm
Thorax circumference (TC) 150,8 mm
Cardiac circumference (CC) 45,6 mm
Femur Length (FL) 30,0 mm
Heart action present. Fetal movements visible. Presentation cephalic II. Amniotic fluid: normal. Cord: 3 vessels.
Placenta: Fundal Posterior, Grannum Grade 0, structure normal.
Fetal anatomy:
The following were visualised and appear normal:
head, brain, face, spine, neck and skin, chest, four-chamber view and great vessels, abdominal wall, gastro-intestinal tract, kidneys and bladder, extremities, skeleton
Echocardiography:
Operator: Dr. Jenkins
Indication for Fetal Echo: Increased nuchal translucency
Image quality: Good
Situs: Normal
Position of stomach: Left sided
Heart size: Normal
Press F12 to save and archive
Press Esc to cancel
User: admin
CAP NUM OVR

Figure 3.5.: Findings System, ViewPoint Report – screenshot of the report modification mask.

- Export medical data of the current examination, for example the VP-Report to CDMS.
- During data entry the current examination is saved periodically.

Available Findings Masks:

- General Reporting
- Prenatal Reporting
- Perinatal Reporting
- Gynaecology Reporting
- Mammogram Reporting
- Ultrasound Reporting
- Endoscopy Reporting
- Echocardiographic Reporting

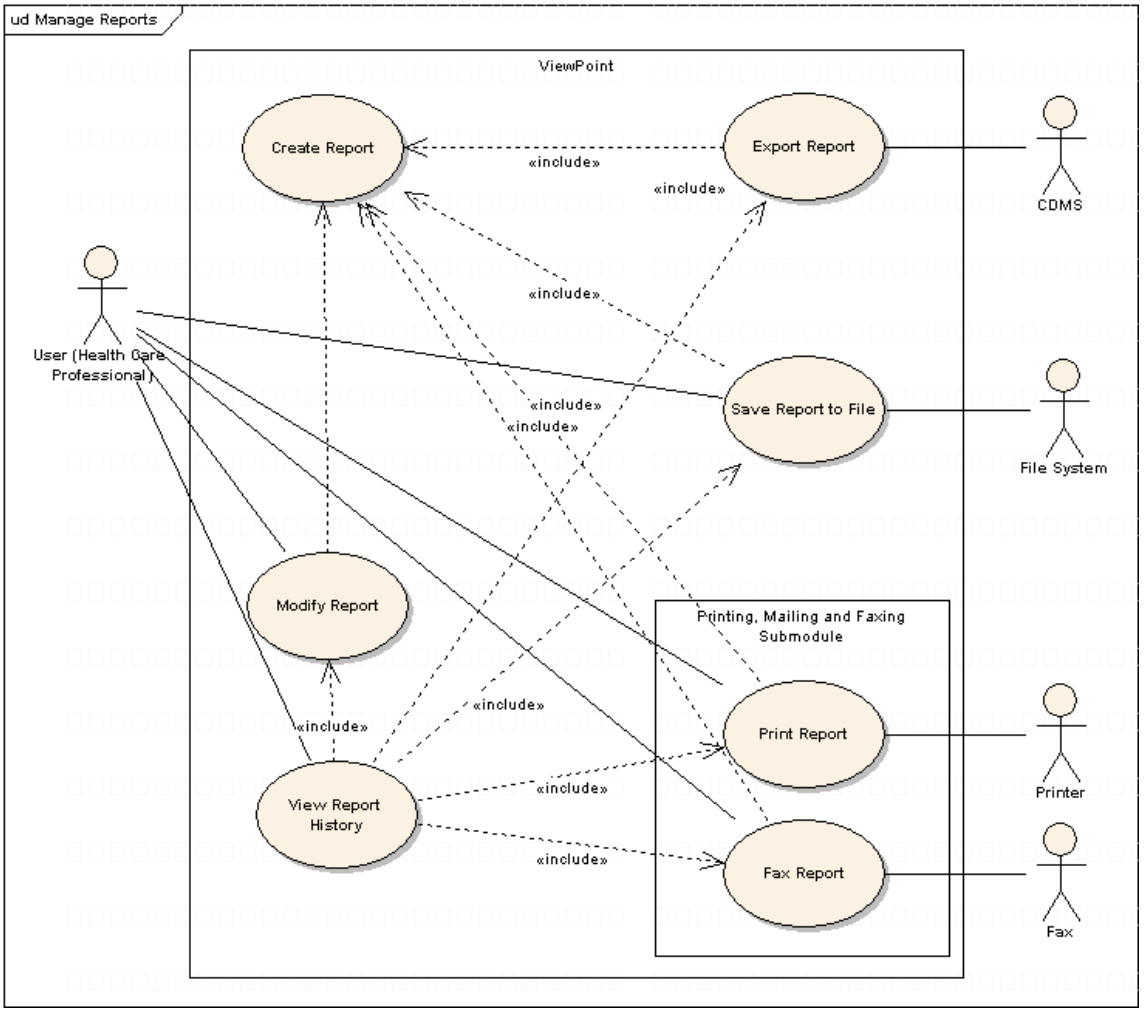


Figure 3.6.: Findings System, ViewPoint Report – UML use case model of the report handling, modified version of the model taken from [Seh05].

Basic Functions:

- Predefined entries in input fields.
- User defined entries in input fields.
- Definition of default values for input fields, which are automatically selected if the corresponding mask is opened.
- Definition of mandatory input fields, which have to be completed to be able to close the related mask and archive the findings.
- Dynamic display of fields, depending on the value of other fields (for example for hiding fields which are not needed).
- Display graphs.
- Generate a report.

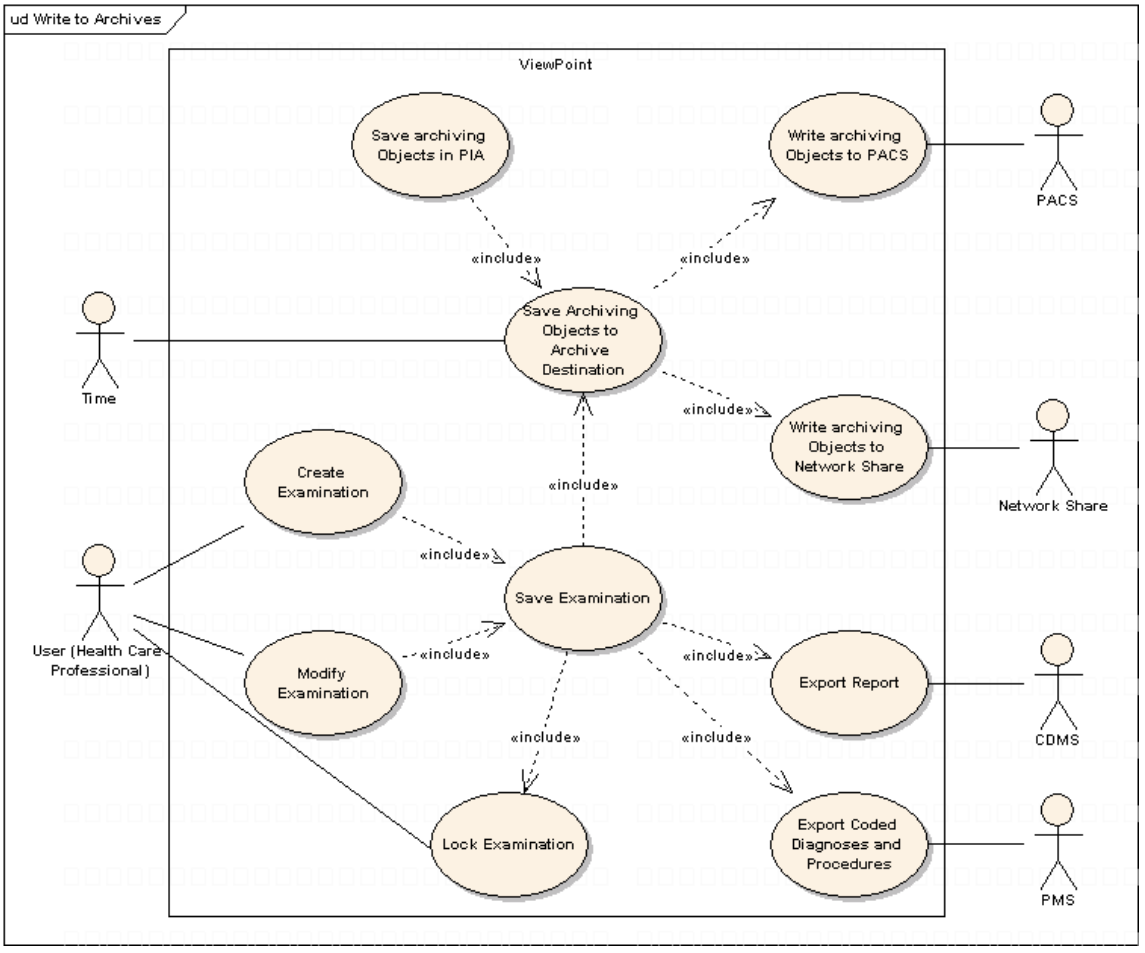


Figure 3.7.: Findings System, Archiving Process – UML use case model, modified version of the model taken from [Seh05]

- Display a report.

Extended Functions:

- Receive and record Ultrasound (US) measurement data as described in 3.3.1.5.
- Manage user defined entries.
- Modify masks with the help of a UI.

3.2.8.3. Image Gallery Submodule

The image gallery of VP is also patient oriented and can be accessed via the examination module. It gives the user extended functions to handle image data. Each examination has a separate set of image data accessible via the image gallery.

If image data should be captured from an analog device, for example an endoscopy device with RGB or S-VHS output, special hardware called Framegrabber is required.

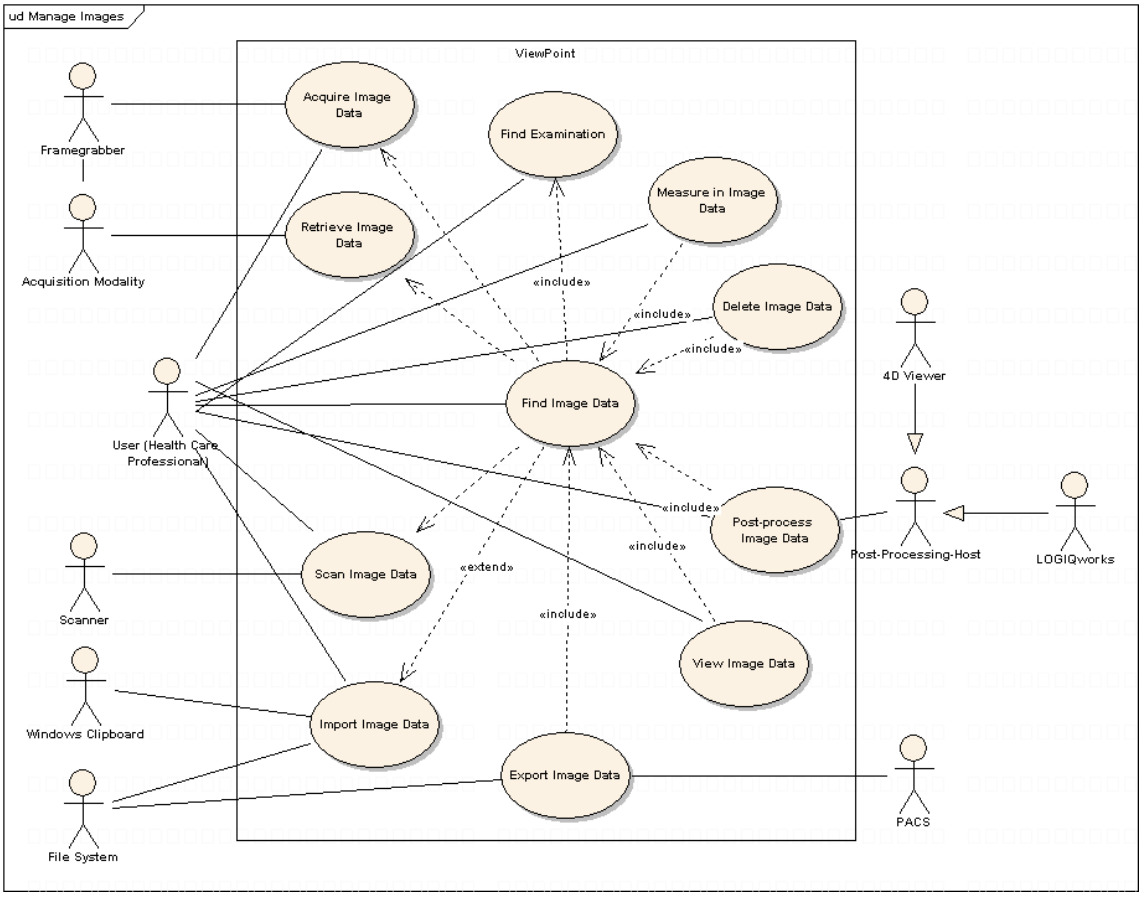


Figure 3.8.: Image Gallery Submodule – UML use cases of image data handling, modified version of the model taken from [Seh05]

Use cases of the image gallery submodule are shown in figure 3.8. Offered functions are:

- View image data.
- Retrieve image data via DICOM interface (see section 3.3.1.2 for details).
- Digitise image data with the help of a Framegrabber. Still pictures and video sequences can be acquired from analog image sources (see section 3.3.1.4 for details).
- Import image data via file import (supported formats are `JPEG`, `PNG`, `TIF` and `BMP`, `AVI` for video sequences).
- Import image data via Microsoft Windows clipboard.
- Import image data via `TWAIN` compatible scanner devices.
- Export image data via file export (supported formats are `JPEG`, `PNG`, `TIF` and `BMP`, `AVI` for video sequences and DICOM).
- Perform measurements on images (for example distances on a US image).

- Post-process image data with:
  - Internal functions like zooming, taking snapshots, commenting and assigning search terms for classification.
  - External functions offered by the LOGIQworks FX and Voluson 4D ViewPlug-ins for processing raw data of the supported ultrasound devices.
    Comment: With the help of these Plug-ins, image editing functions from the related US scanner are made available at the workstation. To a greater or lesser extent, they are separate software packages, which are too complex to be described here and would exceed the scope of this thesis.

3.2.8.4. Printing and Faxing Submodule

The Printing and Faxing Submodule offers functions to print out or to fax a variety of print forms. The print forms range from informal and comprehensive (mostly for internal use, for example internal report for the ward) to official and dedicated (mostly for external use, for example letter to assigning doctor).

The core of nearly every print form is the VP-Report generated by the findings system (see section 3.2.8.2 for details). Beside informal and formal reports, also special print forms are available, for example labels or a partogram. As example an Obstetrical Ultrasound Report is shown in figure 3.9 (page 47) and 3.10 (page 48).

Printers can be connected locally or via the network. Fax functions are provided by the Microsoft Windows Fax Server to which VP connects.

Use cases of the Printing, Mailing and Faxing Submodule are shown in figure 3.6.

Offered functions are:

- Different print forms can be chosen for printing.
- Different print forms can be chosen for mailing.
- Different print forms can be chosen for faxing.
- Availability of print forms can be made dependant on certain conditions (for example specific mask fields have to be completed).
- Assigning doctor is made the standard recipient of the external report (called “letter to doctor”).
- Ability to add graphs to print forms.
- Ability to add other recipients to print forms.
- Ability to add pictures to print forms.

3.2.8.5. Rule-Checking Submodule

The rule-checking submodule is a highly specific submodule which is only employed in the German perinatal reporting module. Here, every birth record has to be exported to the

state's quality assurance institutions within the scope of the German perinatal inquiry at least once per year (see section 3.3.1.10 for details). In order to assure data integrity the institutions also issue rules to reveal integrity errors and to additionally categorise these errors into non-correct but acceptable (so-called soft errors) and non-correct and unacceptable (so-called hard errors).

The rule-checking submodule validates all recorded births and provides the user with a list of all hard and soft errors found within a birth record.

Offered functions are:

- Display a list of all erroneous documented births with the number of errors. It can be configured whether soft and/or hard errors shall be shown.
- Define which birth records are shown in the list. Birth records can be selected by specifying a time span, birth-number or quarter of a year.
- Start checking all birth records of the list or only the currently selected birth record.
- Open examination module of the currently selected birth record (to correct errors).
- Open rule-checking submodule in examination module to validate the currently recorded examination.

3.2.9. System Administration Module (VPAdmin)

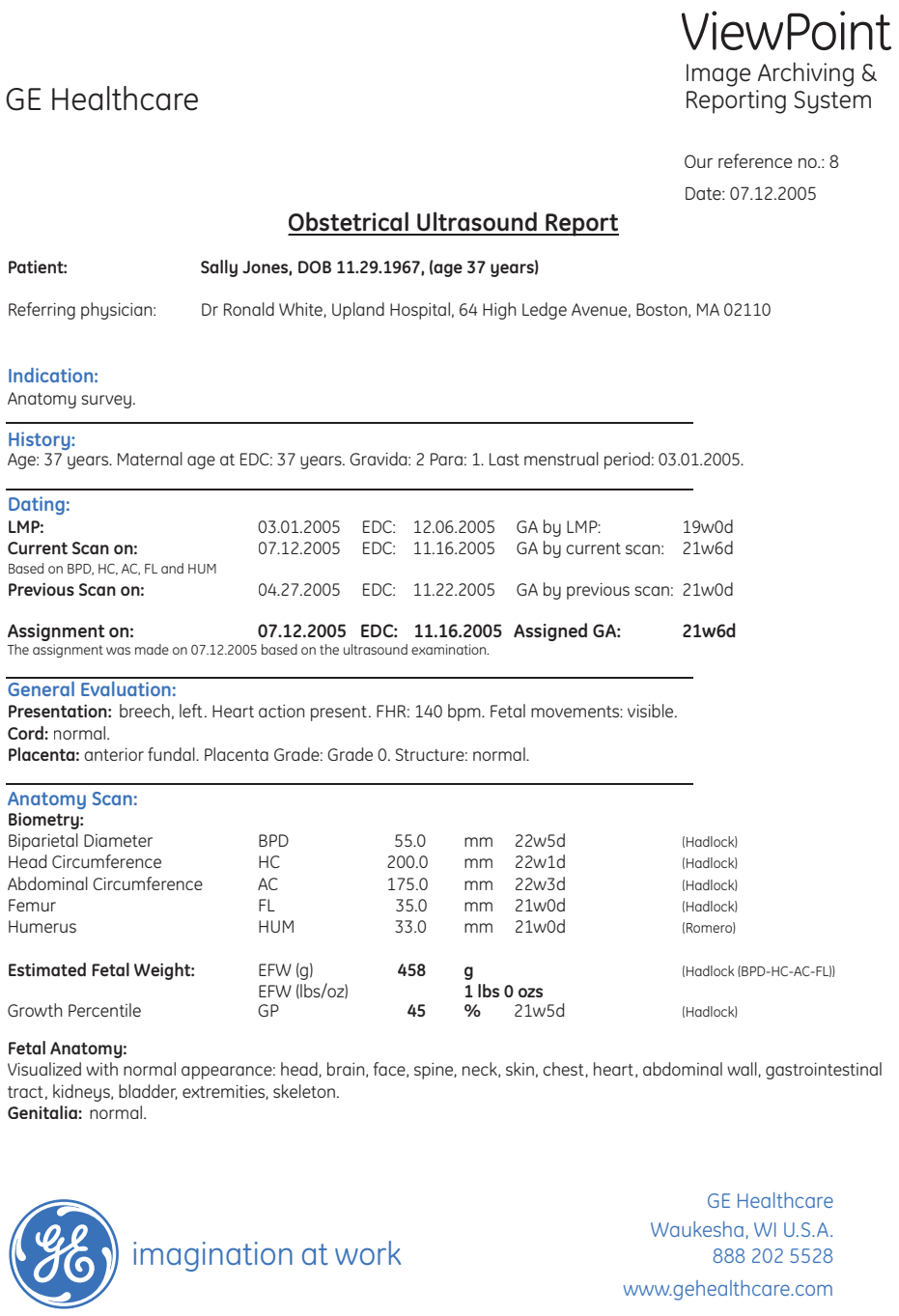
In order to customise VP to the user's needs and to the site's characteristics, it comes with a special administration programme. It is used to setup VP on the server and workstations and to maintain it afterwards.

Part of the configuration is applied to the VP system as a whole (valid for all workstations) and part of the configuration is workstation specific. In order to configure each workstation of a VP network, VPAdmin offers a list view of all installed VP clients together with the workstation specific configuration parameters. For the general setup of the system, a separate configuration mask is available. The common procedure is to set all parameters in the general configuration mask and only if a client needs a different or dedicated parameter adjustment, modifications are made workstation specific.

Emphasising on the functions in detail would not contribute to the goal of this diploma thesis, as the functions of VPAdmin are too complex to be described here. Please refer to [Seh05] or the user manual [Rie05b] for more details.

Vital functions of the VPAdmin to note are the administration of the VP specific user management and the configuration and monitoring of the system's interfaces.

The question of whether the functions of VPAdmin should also be adapted by the Web-based UI will not be answered, as this goes beyond the scope of this diploma thesis.



GE Healthcare

ViewPoint
Image Archiving &
Reporting System

Our reference no.: 8
Date: 07.12.2005

Obstetrical Ultrasound Report

Patient: **Sally Jones, DOB 11.29.1967, (age 37 years)**

Referring physician: Dr Ronald White, Upland Hospital, 64 High Ledge Avenue, Boston, MA 02110

Indication:
Anatomy survey.

History:
Age: 37 years. Maternal age at EDC: 37 years. Gravida: 2 Para: 1. Last menstrual period: 03.01.2005.

Dating:

LMP:	03.01.2005	EDC:	12.06.2005	GA by LMP:	19w0d
Current Scan on:	07.12.2005	EDC:	11.16.2005	GA by current scan:	21w6d
Based on BPD, HC, AC, FL and HUM					
Previous Scan on:	04.27.2005	EDC:	11.22.2005	GA by previous scan:	21w0d
Assignment on:	**07.12.2005**	**EDC:**	**11.16.2005**	**Assigned GA:**	**21w6d**

The assignment was made on 07.12.2005 based on the ultrasound examination.

General Evaluation:
Presentation: breech, left. Heart action present. FHR: 140 bpm. Fetal movements: visible.
Cord: normal.
Placenta: anterior fundal. Placenta Grade: Grade 0. Structure: normal.

Anatomy Scan:
Biometry:

Biparietal Diameter	BPD	55.0	mm	22w5d	(Hadlock)
Head Circumference	HC	200.0	mm	22w1d	(Hadlock)
Abdominal Circumference	AC	175.0	mm	22w3d	(Hadlock)
Femur	FL	35.0	mm	21w0d	(Hadlock)
Humerus	HUM	33.0	mm	21w0d	(Romero)
Estimated Fetal Weight:	EFW (g)	**458**	**g**		(Hadlock (BPD-HC-AC-FL))
	EFW (lbs/oz)		**1 lbs 0 ozs**		
Growth Percentile	GP	**45**	**%**	21w5d	(Hadlock)

Fetal Anatomy:
Visualized with normal appearance: head, brain, face, spine, neck, skin, chest, heart, abdominal wall, gastrointestinal tract, kidneys, bladder, extremities, skeleton.
Genitalia: normal.

imagination at work

GE Healthcare
Waukesha, WI U.S.A.
888 202 5528
www.gehealthcare.com

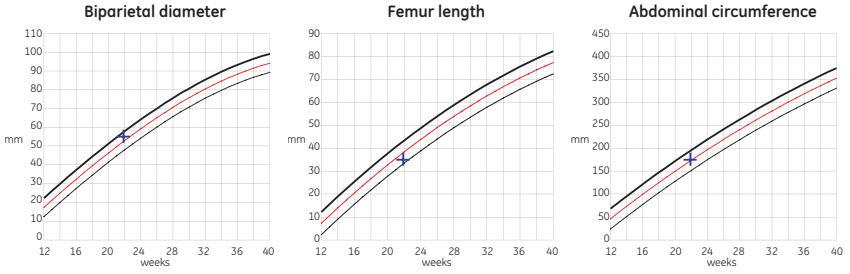
Figure 3.9.: Printing and Faxing Submodule – First side of an obstetrical ultrasound report.

Maternal Structures:

Cervix: normal.
Cervical length: 3 mm.
Left Ovary: normal.
Right Ovary: normal.

Report Summary:

Impression: Growth and anatomy survey appears normal.
Recommendations: We recommend evaluation in 8 weeks



Sonographer: Samantha Jones _____

Physician: Dr. Moore _____

Page 2 of report date 07.12.2005 - Ms Sally Jones, DOB 11.29.1967

Figure 3.10.: Printing and Faxing Submodule – Second side of an obstetrical ultrasound report.

3.3. Current State of Integration

3.3.1. Identification of Available Interfaces

As already stated in the introduction (see section 1.2.2 for details) VP offers different interfaces to other application components or information processing devices to enable data integration. Following the available interfaces will be listed.

3.3.1.1. HL7, HCM, Clinicom, BDT and Other (computer-based)

In the terminology of VP-GE the interface which will be described in this paragraph is called "HIS interface". What is actually meant by this term, is that it is the interface to communicate with the PMS, CDMS or any other application component which is processing with VP redundant data types.

In the typical scenario, patient demographics are imported from the PMS while billing information (like coded diagnoses and procedures) are exported to the PMS. The report text is exported to CDMS. In short, the PMS holds the administrative data hospital-wide, meanwhile the CDMS holds the medical data hospital-wide. As this is the most common configuration, interfaces to other application components using the protocols stated here will not be considered.

VP supports most of the common protocols used for communicating medical and administrative (text) data between the previously mentioned application components: HL7, HCM, clinicom, BDT, `Open DataBase Connectivity (ODBC)` and `SAP-RFC`. Some of the listed protocols are not directly supported by VP. In order to enable their support, a so called gateway software is used. The software product utilised as gateway is Cloverleaf. During import, received messages of a not supported protocol are transformed by the gateway to a VP internal format, which is then further processed in VP. The export works vice versa.

For comparing the different protocols, a distinction between messaging standard and communication technique was drawn. Although in their broadest sense both are protocols according to Holzmann's definition of a protocol [Hol91]. With the term messaging standard a protocol whose main purpose is to communicate complex medical or administrative information independent of the actual transport medium or transport protocol is denoted. Communication technique denotes a protocol whose main purpose is to communicate data through different types of communication links independent of the actual information which is represented by the data.

Supported protocols and the thereby communicated data types are listed subsequently. The third column specifies if VP communicates directly (VP) or indirectly via the gateway Cloverleaf (CL).

Offered functions are:

- Ability to search for and transfer patient demographics of a certain patient from the CDMS to VP via a UI.
- Ability to automatically insert and update all patient demographics which are stored in the CDMS into VP.

Table 3.1.: Overview of the interfaces of ViewPoint to CDMS and PMS.

Messag. Standard	Comm. technique	Comm. partner	Import	Export
HL7	file transfer	CL	1,2	1,3,4
HL7	socket	VP	1,2	1,3,4
HCM	socket, file transfer	CL	1,2	1,3,4
clinicom	socket, file transfer	CL	1,2	1,3,4
-	`ODBC`	VP	1	-
-	`SAP-RFC`	VP	1	-
BDT	file transfer	VP	1	-

Legend
CL: Cloverleaf (indirect communication)
VP: ViewPoint (direct communication)
1: patient demographics
2: clinical orders
3: findings report
4: revenue codes like for example ICD-10 or OPS

- Ability to process orders via a UI (see section 3.2.8 for details).
- Ability to automatically export coded diagnoses and procedures of an examination record, when it is archived.

3.3.1.2. DICOM (computer-based)

Many types of the DICOM standard are supported by VP. They will be described subsequently at an quite abstract level. For details please refer to DICOM conformance statement of VP [Rus06].

General PACS Functionality - Storage Concept In its function as PACS, VP can send and receive image data via the DICOM standard and can store it in its archive. Advanced functions like query and retrieve functions, in which other modalities or PACS request image data from VP, are not supported.

The received or acquired image data is stored in the VP archive using the DICOM file format. VP can act as a long-term storage archive, in which image data is stored as long as it has to be kept according to legal regulations (shown as alternative A in figure 3.11). If the hospital has a central PACS for long-term storage, VP stores the image data only temporarily for reviewing and post-processing purposes in its archive and forwards all image data immediately after reception/acquisition via DICOM to the central archive (shown as alternative B in figure 3.11). In this case image data is deleted from the VP archive after a specific amount of time, but can be retrieved later on from the central archive via DICOM.

DICOM image data VP offers functions to handle DICOM image data. Speaking in the context of DICOM, image data comprise still images (so-called single frame) and video

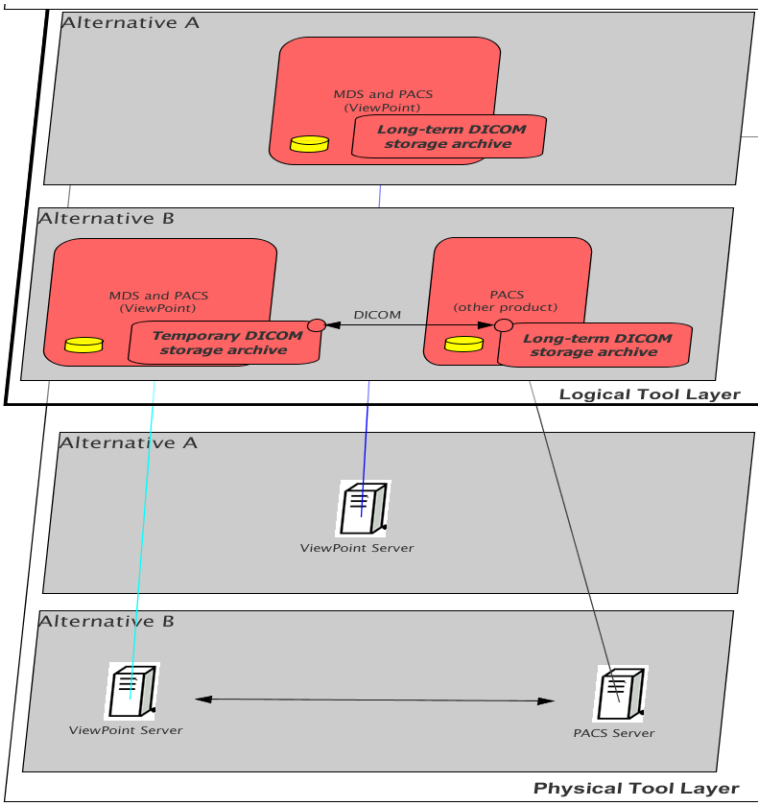


Figure 3.11.: Logical and physical tool layer of the 3LGM^2 model showing the two alternatives of the DICOM storage concept of ViewPoint.

sequences (so-called multi frames). One special characteristic of DICOM is that all image data of an examination of a certain patient is collected in a virtual container called "study". In case VP receives or sends DICOM data over the network it utilises this container. In other words, if VP, for example, receives image data from an imaging modality, it maps the DICOM study containing all image data to a VP examination, so that it is accessible via the image gallery submodule (see section 3.2.8.3 for details).

Offered functions are:

- Receive image data and give the user the opportunity to handle it adequately via an UI (for example assign a study to an examination record).
- Send image data.
- Archive image data in VP for short term storage.
- Archive image data by sending it, for example, to PACS for long-term storage.
- Handle multiple DICOM sources and destinations.

DICOM Worklist With the utilisation of DICOM worklists, it is possible to receive patient demographics at the US device from VP. This procedure works as follows. A patient record is opened (or created) in VP. The sonographer initiates the retrieval at the

US device, which sends a request to the workstation on which VP runs. VP answers this request by sending the related patient demographics to the US device.

Offered functions are:

- Ability to configure one of the following sources for creating a worklist:
  - Current patient
  - Appointments
  - Orders
- Ability to map one order from the PMS to exactly one worklist entry.

DICOM SR DICOM SR is a way to send US measurement data by using the DICOM protocol. US measurement values are exchanged between the imaging modality and VP like image data. Measurement data is processed as described in 3.3.1.5.

3.3.1.3. CORBA (computer-based)

Since Version 5, VP is based on the `Common Object Request Broker Architecture (CORBA)` architecture. At the moment the by such a kind of architecture offered features are only available to VP clients and servers. The access of external application components to `CORBA` based services is not supported yet.

3.3.1.4. Analog Image / Video Capture (computer-based)

In order to capture images or video sequences from imaging modalities which offer analog video output only, a Framegrabber card is used. With the help of this special hardware, VP converts analog video signals into digital ones which can then be processed like standard DICOM data. The capturing process can be initiated by the utilisation of a switch, which has to be connected to the capturing workstation.

3.3.1.5. US Measurement Data (computer-based)

In the common work flow of an US examination, all biometric measurements are performed directly at the US device and, therefore, stored on it. In order to prevent duplicate data entry, additional work connected to it and error probability, electronic transfer of most measurement values from the US device to VP is supported. The transfered values are automatically inserted into the findings masks of the currently opened patient record.

Offered functions are:

- Measurement data transfer by the utilisation of the following protocols/connection types:
  - Serial cable connection.
  - Parallel cable connection.

  - DICOM SR (see section 3.3.1.2 for details).
  - Microsoft Windows network share.

- Customisation of the transfered values, for example when the labels of the sent values differ on account of country specific firmware versions of the US device.

3.3.1.6. Trium CTG Online Monitoring (computer-based)

Trium CTG Online Monitoring is a Web-based cardiotocograph monitoring system. VP offers an interface to export patient case information to this specific CTG monitoring system via a proprietary interface (see section 6.2.3 for more details).

3.3.1.7. Chip Card Reader (computer-based)

VP offers an interface to import patient demographics and health insurance information from the German health insurance card. Different chip card readers are supported, connected via a serial cable or integrated in a keyboard.

3.3.1.8. Printing and Faxing Interface (paper-based)

As described in section 3.2.8.4, VP offers functions to print out or fax different variations of the VP-Report and other information.

3.3.1.9. Scanning Interface (paper-based)

As described in section 3.2.8.3, it is also possible to digitise paper-based documents with the help of a `TWAIN` scanner.

3.3.1.10. Other

German Perinatal Inquiry (computer-based) As mentioned above, this interface is only employed in the German perinatal reporting module. Within the scope of the German perinatal inquiry, data of all births in Germany is collected by central quality insurance institutions. As VP is also an application component with which births can be documented, it offers an interface to support this inquiry adequately.

Offered functions are:

- Export birth data in the respective coding (for example ICD-10 etc.) to be send to the quality assurance institutions. After the export the exported examinations will be locked.
- Configure the hospitals identification data (which is included in the exported birth data to uniquely identify the origin of the data).
- For more functions please refer to section 3.2.8.5, as the rule-checking submodule works closely related to the interface for the German perinatal inquiry.

Fetal Medicine Foundation London (computer-based and paper-based) Since decades the FMF is specialised in assessing the risk of chromosomal abnormalities which are the major cause of perinatal death and childhood handicap. The risk is determined by statistical analysis of biochemical and ultrasound markers acquired during screening tests. In order to improve the accuracy of the screening tests, the FMF collects data of all performed screening tests in which the risk of chromosomal abnormalities was assessed as well as their outcome.

In order to support FMF in their work, VP offers an interface to easily gather and export the data to the FMF.

Offered functions are:

- Export FMF relevant data to file.
- Send FMF relevant data via email to the FMF.
- Display data which will be exported, customise view depending on the examiner.

3.3.2. Model of Current State of Integration

The current integration degree of VP within the HIS has been modelled by using the 3LGM^2 Meta model (see section 2.3.2 for details). In figure 3.12 (page 57, for full-page view please see figure A.1 in the appendix) the logical tool layer and in figure 3.13 (page 58, for full-page view please see figure A.2 in the appendix) the physical tool layer of the model is shown.

The domain layer of the model was created based on the reference model proposed by Hübner-Bloder et al. [HBABW05]. It is not shown as it is irrelevant to the objectives of this thesis. By offering the Web client no changes in the enterprise functions which are supported by VP will occur. Only the way how data stored in VP is accessed will alter. Thereby the domain layer will not change. Please refer to table A.1 on page 121 for an overview of which enterprise functions are supported by VP.[1]

3.4. Weak Point Identification

In the first part of this section all weak points of the current solution will be identified. Subsection 3.4.1 will focus on certain requirements and missing features which are weaknesses of the Native client. Subsection 3.4.2 will concentrate on weak points of the Native client in respect to its integration potentials.

In the second part of this section all weak points of the future solution will be identified. Web-based technology also comes with certain disadvantages or critical issues which have to be considered carefully during requirement specification and the later implementation stage. They will be identified in subsection 3.4.3.1. Moreover, certain constraints are set by VP and its special characteristics. They will have to be preserved by the future Web client. A description of these constraints can be found in subsection 3.4.3.2.

[1] Until now only a German version of this model exists.

3.4.1. Weak Points of the Native VP Client

The points given here are derived from the description of VP given in section 3.2.

As only Microsoft Windows 2000, XP or 2003 Server with certain Service Packs are supported (see section 3.2.1 for details), VP cannot run on older Microsoft Windows versions or versions with different (non-supported) Service Packs. Neither can it run on other Platforms like Linux or Apple Macintosh which may also be present in an clinical infrastructure.

WP 1: Incompatibility with other Platforms and Microsoft Windows versions with different Service Packs.

VP requires certain software and hardware equipment (see section 3.2.1 for details). Although not the most recent technology is needed, still the requirements of VP usually lead to the need of upgrading the workstation, as most computer systems at hospitals are out-dated. This is caused by the comparatively long lifespan of computer systems at hospitals, which are not replaced by newer (better) systems due to budget limitations and organisational constraints.

WP 2: Comparatively high hardware requirements.

VP has a high bandwidth utilisation (see section 3.2.1 for details) and due to CORBA (see section 3.3.1.3 for details) requires a stable network connection. These network characteristics are in most cases only fulfilled in the wired LAN of a hospital. So the usage of the Native client is not possible from sites like remote places which only have low bandwidth connections to the VP server. In addition, usage of the Native client over unstable networks like Wireless Local Area Network (WLAN) (inside the hospital) is critical, as the IP address of a client might change during roaming inside the WLAN. Also, sudden network interruption may occur.

WP 3: High bandwidth required. Access from remote sides nearly impossible.

WP 4: Stable network connection required. Usage over WLAN nearly impossible.

In order to be able to access VP from a workstation, manual installation and configuration of the Native client is required as described in section 3.2.1. Moreover, after setup each VP client has to be maintained during its lifespan in a hospital, resulting in additional efforts.

WP 5: Manual installation and configuration required to run VP on a workstation.

WP 6: Maintenance efforts for each client.

As stated in section 3.2.3 the findings and image data stored within VP only have medical significance in combination. So considering the rich and detailed documentation features, the broad connectivity to diagnostic devices and the handling of image data in a patient-case-examination oriented way, VP is the preferred application component at the functional diagnostic unit.

WP 7: Dependency on VP.

VP has and requires a user management of its own. Each user who wants to access VP has to log in, assuming she/he has a valid user account in VP and did not yet login during the current session.

WP 8: No smooth switching between Native client and other application components on a workstation.

As the sales team reported, in most tenderings a Web client is demanded. VP does not offer a Web-based UI. Therefore, major orders may be lost due to this missing feature.

WP 9: No Web client offered for VP.

3.4.2. Weak Points in Respect to Integration Potentials of the Native VP Client

The points given here are derived from the integration analysis of VP given in section 3.3 and the description of VP given in section 3.2. They are classified by type of integration.

3.4.2.1. Data Integration

To a high degree VP does already support data integration (see section 3.3.1 for details). For example, patient demographics are imported from the PMS, coded diagnoses and procedures are exported to the PMS, VP-Reports are exported to the CDMS, image data is exported to the PACS and biometric measurement data is transfered from medical devices to VP. These interfaces along with other application components are illustrated in figure 3.12.

A weak point concerning data integration is, that no direct referencing between VP-Reports and image data is possible as soon as they are exported to other application components. That means that in case an exported VP-Report is reviewed in CDMS, no direct access to the related image data is possible, as the VP-Report does not contain any references/links to those image data stored in the PACS. So if the user also wants to view the image data she/he has to access the PACS and search for the related image data by specifying the patient identity.

WP 10: No full data integration guaranteed as links/references between VP-Report and image data are not exported.

Other weak points concerning data integration are that data transfer from/to certain diagnostic devices or other application components is not (adequately) supported. As the

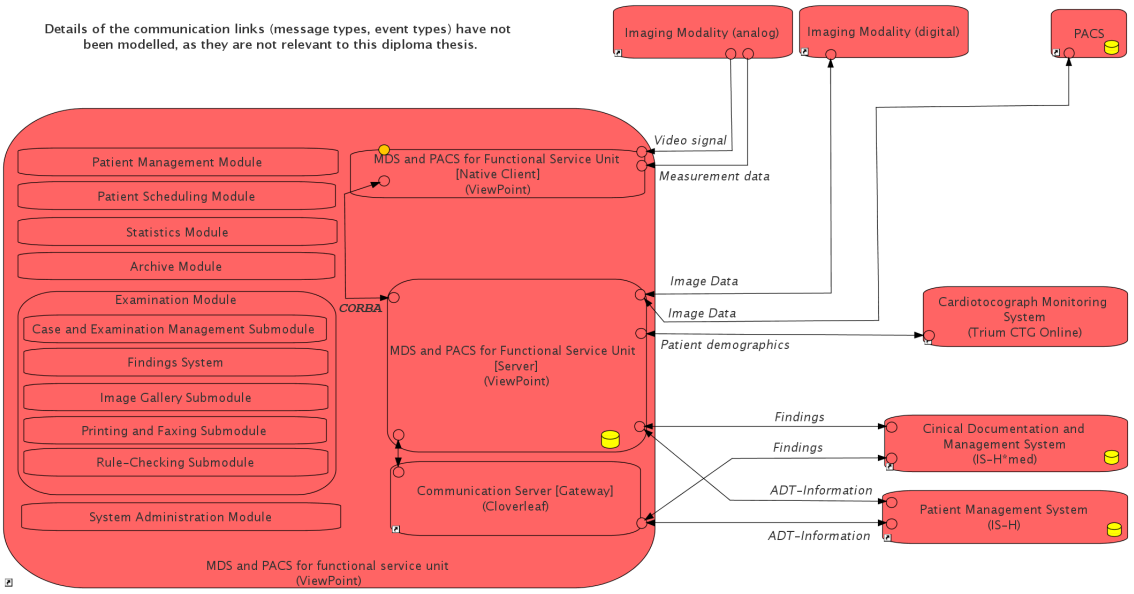


Figure 3.12.: Logical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system.

extension of an existing or the creation of a new data transfer interface is not within the scope of this thesis, it will not be analysed in detail and no requirements will be derived from it.

3.4.2.2. Access Integration

As described above, it is common that VP is only installed on the functional diagnostic unit. At the ward, at other departments or from remote sites like the assigned physician's practice or the consulted specialist's hospital, where information stored in VP is consulted, direct access to VP is at the moment not possible.

There are many reasons for this circumstance. Most of them have been covered in section 3.4.1. WP 1 to WP 6 already limit the number of potential workstations on which the Native client could and would be installed. Further limitations are set by organisational constraints. For example, locations outside the functional diagnostic unit may belong to other organisational units which have a separate budget and a different set of priorities. Thus, instead of adding a the Native client of VP to their IT infrastructure they might prefer to invest in things from which they could gain more of a direct benefit.

It can be summarised that the Native client is inadequate to provide access to VP wherever it is needed and no other means for this purpose are offered by VP, for example a Web client (WP 9).

So to provide the information needed at locations where access to VP is not possible, the VP-Report and related image data have to be printed out (see section 3.3.1.8 for details) and transfered. According to [HAWB04, 160] this is a transcription of data and leads to so called media cracks. Apart from the typical problems which go along with the duplication of data, like data inconsistency, this procedure is also quite time-consuming.

Time-consuming in so far as the transfer of the printouts via mail is concerned (internal or external dependent on where the receiver is) which takes considerably longer than electronic transfer. Transfer via mail takes considerably longer than electronic transfer. This weak point is illustrated by figure 3.13.

WP 11: No access integration guaranteed causing transcription of data and media cracks.

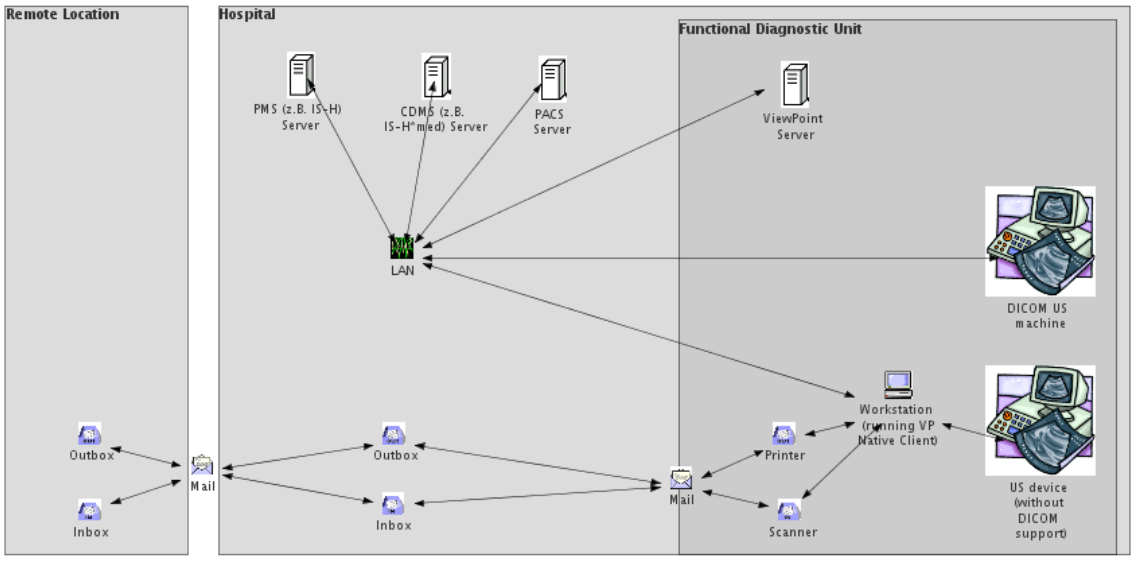


Figure 3.13.: Physical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system.

3.4.2.3. Presentation Integration

The UIs of software products being employed in the clinical environment differ greatly. Even products of the same manufacturer have quite often a different UI. As no attempts to standardise the UIs of software products in the clinical environment exist, it is impossible to create a common UI layout and operation.

Within the VP family presentation integration is guaranteed as every reporting module has the same UI. So a user who is familiar with one reporting module can handle/operate another reporting module intuitively.

WP 12: User expects the same appearance of all products within the VP product family.

At obstetrical departments VP is often sold in combination with Trium CTG Online as an all-in-one solution. Trium Analysis Online GmbH is a cooperate partner of VP-GE. As Trium CTG Online is Web-based and was not originally designed to be sold as an enhancement to VP, its UI is quite different to the one of VP.

WP 13: No presentation integration between VP and Trium CTG Online.

3.4.2.4. Contextual Integration

VP offers no means to synchronise

- user information (like login and password) and
- patient identification information (like name or hospital ID)

with other application components running on one workstation. Although a standard (CCOW) and various methods (like integrated Microsoft Windows login) exist, VP supports non of them and, thus, does not guarantee contextual integration.

WP 14: No means for contextual integration provided.

WP 15: CCOW not popular. (Only few manufacturers implemented it.)

3.4.3. Weak Points of a Web Client

3.4.3.1. Weak Points of a Web Client in General

With the utilisation of a Web interface, everyone connected to the Web server has access or can at least try to gain access to the information provided thereby. The chances of unauthorised access are higher in such an environment, as the only tool needed to access the Web-based UI is a Web browser. So compared to a Native client, in which special software has to be installed to gain access to the information stored within the system, a Web-based UI offers even to novices a noticeable chance to break into the system.

In the LAN of the hospital it can be assumed that all participants of the network are authorised or will at least not attempt to break into the system. As soon as the Web-based UI is made accessible via the Internet, unauthorised access attempts or other types of attacks on the security of the system must be assumed.

WP 16: High risk of security threats.

The operation of Web-based clients or in general of Web-pages tend to be slow compared to Native clients (see section 2.6.2 for details).

WP 17: Limited performance.

High chance of data loss if connection is interrupted (network cable accidentally unplugged). A Native client offers better means to protect against data loss in case of network interruptions.

WP 18: Higher risk of data loss.

Not every required function may be realisable through a standard Web page (see section 2.6.2 for details).

WP 19: Limited functionality.

A variety of Web browsers is available on the market, only few of which are popular. Although Web browsers like Microsoft Internet Explorer or Firefox state that they support many Internet standards, the actual recognition and implementation of the corresponding standard is Web browser dependent and may even depend on the Platform on which the Web browser is used (see section 2.6.2 for details).

WP 20: Web browser specific implementation of the Web standards.

The default functions offered by a Web browser are usually not sufficient. Plug-ins or Helper Applications are required to realise the missing functions. Most of the Web browser extensions are Platform- or even Web browser-dependent and, therefore, destroy the advantage of the Web client to be Platform- and Web browser-independent (see section 2.6.2 for details).

WP 21: Plug-ins or Helper Applications are Platform and Web browser specific.

Plug-ins or Helper Applications need to be installed on the workstation (see section 2.6.2 for details). New versions are also issued for Plug-ins or Helper Applications, so once installed they also have to be maintained to support the version which is required by the Web client.

WP 22: Manual installation required to use Plug-ins or Helper Applications on a workstation.

WP 23: Maintenance efforts for installed Plug-ins or Helper Applications on a workstation.

On account of the Web browser and Platform dependency it is indispensable to test the Web client for each Web browser and on each Platform which should be supported. In case of malfunction or insufficient performance the Web client has to be optimised for each Web browser an Platform.

WP 24: Testing and optimisation efforts required to support different Web browsers on different Platforms.

WP 25: Efforts increase with the number of Web browsers and Platforms which are supposed to be supported.

Every Web browser offers means to limit its features in order to protect it against security threats. Those features are, for example, acceptance of Cookies, allowance of Pop-Ups, allowance of execution of Java Script etc.. Thus, the default configuration of a Web browser may prevent the usage of features utilised by the Web client. The default configuration has to be modified to be able to use the Web client.

WP 26: Modification of the default Web browser configuration required.

Moreover, certain Web browser extensions may utilise other network ports than the standard HTTP port. If the client and the server are separated by a Firewall, additionally needed ports have to be activated. This is especially relevant for access to VP from remote sites.

WP 27: Non-standard ports required by Web browser extension may be disabled by a Firewall.

3.4.3.2. Points to Be Considered for a Web Client as an Alternative UI to VP

The UI of a Web client for VP has to be defined. The UI definition of the Native client is in proprietary language (see section 3.2.8.2 for details) and cannot be used instantly to define the Web-based UI. So in the worst case two UI definitions exist, which keep redundant data and have to be maintained if changes in their content occur.

WP 28: High risk of inconsistency if the UIs of the Native client and the Web client are defined separately.

WP 29: Additional maintenance efforts required if the UIs of the Native client and the Web client are defined separately.

VP owns image data whose quality, integrity and medical significance have to be preserved.

WP 30: Critical image data to be handled appropriately.

VP-GE quality assurance and fast version release rely strongly on automatic testing. Before the official version is released most tests are performed automatically by test tools.

WP 31: UI of VP is tested automatically by scripts and other testing tools.

The Native client of VP interacts with a variety of hardware devices. The following listing shows the corresponding devices and the used hardware interface to control and access it:

- Printer (parallel or USB port) as described in section 3.2.8.4.
- Chip card reader (serial or USB port) as described in section 3.2.4 and 3.3.1.8.
- Framegrabber (PCI slot) as described in section 3.2.8.3 and 3.3.1.4.
- Scanner (USB port) as described in section as described in section 3.3.1.9.
- Ultrasound device (serial port) for measurement data transfer as described in section 3.2.8.2 and 3.3.1.5.

WP 32: Means to control/access hardware required.

4. Definition of Requirements

4.1. Outline

In the preceding chapter the current functions and the current degree of integration of the Native client have been analysed. Furthermore, weak points of the Native client and potential weak points of the future Web client have been identified.

The aim of this chapter is to create a general requirement definition. It will be derived from the previously identified weak points in section 4.2. Based on this definition the future degree of integration will be modelled in section 4.3.

The succeeding chapter will then specify the generally defined requirements in more detail.

4.2. Requirement Definition

I. General requirements

RD 1.1: VP shall provide a Web client.

Rationale: Elimination of WP 9.

RD 1.2: The Web client shall provide all functions of a Native client as specified by the requirements specification [Seh05]. If this is not possible or feasible at once, it shall be implemented in an incremental approach or it shall provide at least the functions which are demanded by the customers. In case the incremental approach is chosen, features should be prioritised according to their importance to the customer and added stepwise in each new version release.

Rationale: The Web client should be an equivalent alternative to the Native client of VP. Implementation of all features may not be possible or feasible at once due to lack of resources like money, time and man power.

RD 1.3: The Web client shall provide access to VP from all locations where access to VP is required.

Rationale: Elimination of WP 11 and WP 7. VP is the preferred solution to access VP-Report and the related image data at once. The fulfilment of this requirement is also a suitable workaround for WP 10, as the data can be directly accessed through VP and does not have to be exported to other application components to be available.

II. Requirements concerning supported Platforms, Web browsers and required Web browser extenions

RD 2.1: The Web client shall support as few Platforms and OS versions like different Service Packs as are needed to fulfil RD 1.3.

Rationale: Elimination of WP 1, WP 21, WP 24 and WP 25. By keeping the number of supported Platforms and OS versions as low as possible, while still supporting all required Platforms and OS versions, the testing and optimisation efforts can be kept to a minimum. This will also increase the number of applicable Web browser extensions.

RD 2.2: The Web client shall support as few Web browsers as are needed to fulfil RD 1.3.

Rationale: Elimination of WP 20, WP 21, WP 24 and WP 25. By keeping the number of supported Web browsers as low as possible, while still supporting all required Web browsers, the testing and optimisation efforts can be kept to a minimum. This will also increase the number of applicable Web browser extensions.

RD 2.3: The Web client shall utilise as few Plug-ins or Helper Applications as are needed to fulfil RD 1.2.

Rationale: Elimination of WP 19, WP 22 and WP 21. By keeping the number of utilised Plug-ins or Helper Applications to a minimum, the corresponding installation efforts are reduced.

III. Requirements concerning quality of supported networks

RD 3.1: The Web client shall support low bandwidth connections to fulfil RD 1.3.

Rationale: Elimination of WP 3. Clients which are connected to the server of VP via a low bandwidth, for example at remote places, should be supported.

RD 3.2: The Web client shall also support unstable networks to fulfil RD 1.3.

Rationale: Elimination of WP 4. Clients which are connected to the server of VP via an unstable network like WLAN devices should be supported.

RD 3.3: The Web client shall perform and respond in such a manner that the user is not hindered in doing her/his actual work.

Rationale: Elimination of WP 17. The operation of a Web client is generally slower than the operation of a Native client. Still, with optimisation a sufficient performance could be achieved while fulfilling RD 3.1 and RD 3.2.

RD 3.4: The Web client shall utilise only `HTTP` port 80 or `Hypertext Transfer Protocol Secure (HTTPS)` port 443.

Rationale: Elimination of WP 27. Utilisation of the Web client should also be possible if the server is protected by a Firewall. `HTTP` and `HTTPS` ports are the standard ports and commonly forwarded by a Firewall. Due to a restrictive security policy of the hospital it may be impossible to activate other ports.

IV. Requirements concerning data security

RD 4.1: The Web client shall assure secrecy and confidentiality.
Rationale: Elimination of WP 16. Access via the Web client to the data stored within VP must only be available to authorised users and must be protected against unauthorised access [Tan03, 722].

RD 4.2: The Web client shall offer sufficient means for authentication.
Rationale: Elimination of WP 16. It must be assured that each user logged into the Web client is in fact the user she/he claims to be [Tan03, 722].

RD 4.3: The Web client shall offer sufficient means for non-repudiation and accountability.
Rationale: Elimination of WP 16. It must be assured that it can be unambiguously tracked which user made which modifications of data via the Web client [Tan03, 722]. Additionally, even read-only access via the Web client must be tracked.

RD 4.4: The Web client shall offer sufficient means for integrity control.
Rationale: Elimination of WP 16. It must be assured that during communication between VP server and Web client only the original message as sent by the source is received by the destination and not a maliciously modified one [Tan03, 722].

RD 4.5: The Web client shall offer means to protect against data loss.
Rationale: Elimination of WP 18. In case of malfunction of the workstation on which the Web client is running or in case of an interruption or congestion of the network connection to the server, it must be assured that no data entered or modified via the Web client is lost.

V. Requirements concerning installation and maintenance

RD 5.1: The Web client shall require no or at least a minimum of installation and configuration efforts. (Coherence with RD 2.3)
Rationale: Elimination of WP 5, WP 26 and WP 22.

RD 5.2: The Web client shall require no or at least a minimum of maintenance efforts. (Coherence with RD 2.3)
Rationale: Elimination of WP 6 and WP 23

VI. Requirements concerning interaction with other application components

RD 6.1: The Web client shall offer means to switch from itself to another application components running on the same workstation without having to reenter login information.

RD 6.2: The Web client shall offer means to switch from another application components running on the same workstation to itself without having to reenter login information.

RD 6.3: The Web client shall offer means to switch from itself to another application components running on the same workstation without having to reenter patient identification information to preserve the patient context.

RD 6.4: The Web client shall offer means to switch from another application components running on the same workstation to itself without having to reenter patient identification information to preserve the patient context.

Rationale: Elimination of WP 14 and WP 8.

VII. Requirements concerning the UI

RD 7.1: The UI of the Web client shall be based on the existing masks definition if possible and feasible (see 3.2.8.2 for details). If not possible and feasible, the Web client shall utilise a standardised UI definition language so that the UI of the Native client and the UI of the Web client can be created from a single definition.
Rationale: Elimination of WP 28 and WP 29.

RD 7.2: The Web client shall offer a UI which resembles the one of the Native client in appearance and is controlled similarly.
Rationale: Elimination of WP 12.

RD 7.3: The UI of the Web client shall offer more similarity to the UI of Trium CTG Online than the UI of the Native client and should thereby point out the close association between both products.
Rationale: Elimination of WP 13.

RD 7.4: The Web client shall offer a UI which is compatible to automatic testing.
Rationale: Elimination of WP 31.

RD 7.5: The Web client shall preserve the quality and integrity of all image data provided to the user via the Web-based UI.
Rationale: Elimination of WP 30.

VIII. Requirements concerning hardware

RD 8.1: The Web client shall keep its hardware requirements to a minimum.
Rationale: Elimination of WP 2.

RD 8.2: If possible the Web client shall offer means to control to VP connectable hardware devices.
Rationale: Elimination of WP 32.

4.3. Future State of Integration

The future degree of integration of VP within the HIS has been modelled by using the 3LGM^2 Meta model (see section 2.3.2 for details). The model of the future degree of integration VP is based on the following assumptions:

- The Web client is realised based on the requirements defined in section 4.2.
- RD 6.1, RD 6.2, RD 6.3 and RD 6.4 are realised by the utilisation of CCOW.

- The Web client is utilised wherever access to data stored in VP is needed and was provided on a paper-based basis.
- The Web client is an alternative UI to the Native client but no replacement for it.

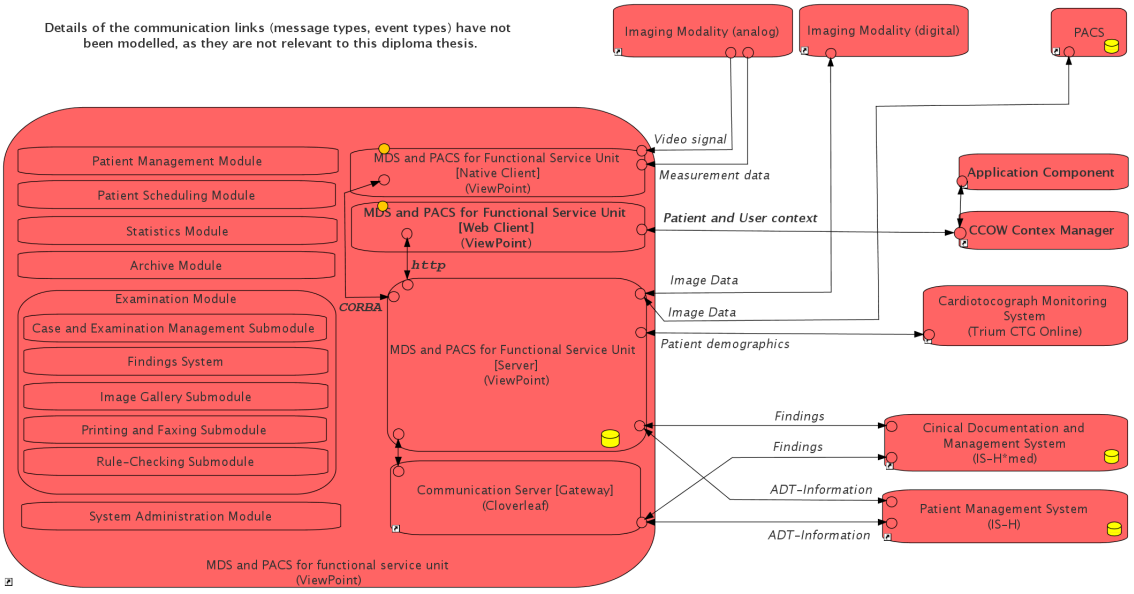


Figure 4.1.: Logical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system.

In figure 4.1 (for full-page view please see figure A.3 in the appendix) the logical tool layer of the model is shown. The newly added elements are labelled with bold letters. The difference to the model of the current degree of integration (as shown in figure 3.12) is that VP now also offers a Web client which communicates with the VP server via HTTP. Moreover, the Web client provides a CCOW interface with which it is possible to synchronise patient and user context with other application components via a CCOW context manager. The component denoted as "application component" is a placeholder for any application component(s), which support(s) CCOW and run(s) on the same workstation as the Web client of VP. It is vital to note that no other interface is affected by the Web client.

In figure 4.2 (for full-page view please see figure A.4 in the appendix) the physical tool layer of the model is shown. The newly added elements are labelled with bold letters. The difference to the model of the current degree of integration (as shown in figure 3.13) is that now access to data stored in VP is provided via physical data processing components from every location. Paper-based distribution of the required data is no longer needed.

The domain layer of the model was created based on the reference model proposed by Hübner-Bloder et al. [HBABW05]. It is not shown as it is irrelevant to the objectives of this thesis. By offering the Web client no changes in the enterprise functions which are supported by VP will occur. Only the way how data stored in VP is accessed will alter. Thereby the domaine layer will not change. Please refer to table A.1 on page 121 for an overview of which enterprise functions are supported by VP.[1]

[1] Until now only a German version of this model exists.

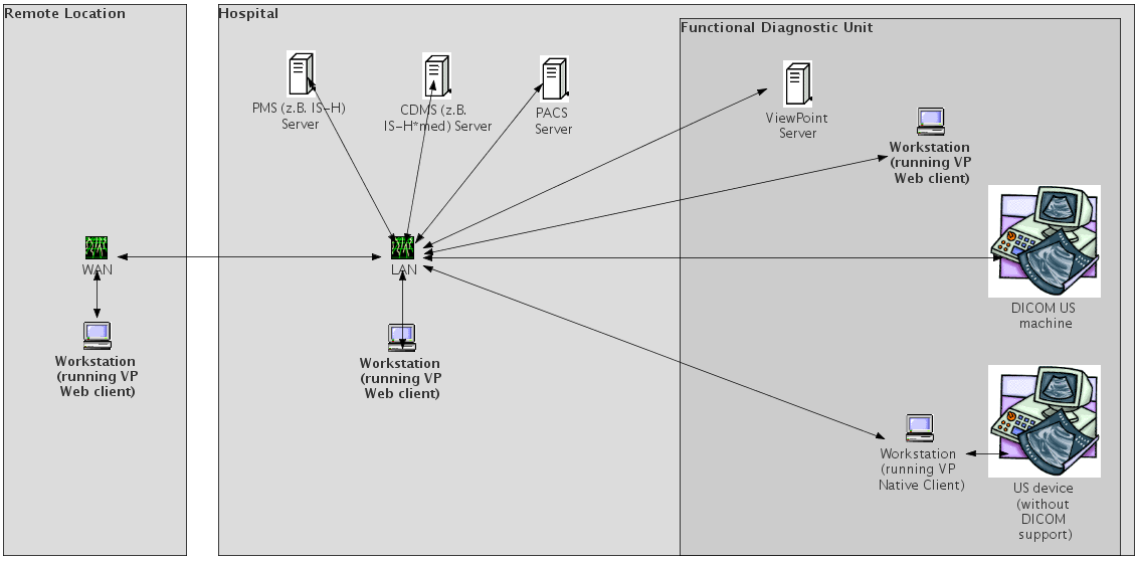


Figure 4.2.: Physical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system.

5. Specification of Requirements

5.1. Outline

In the preceding chapter the requirements for a Web client have been defined in general.

The aim of this chapter is to define these requirements in more detail wherever possible. In section 5.2 issues which are yet unknown and need to be researched to be able to specify certain requirements in more detail will be discussed. The actual requirement specification will be set up in section 5.3.

The succeeding chapter will analyse and compare different products and an Terminal server as alternative technology to the Web-based UI on basis of the requirement specification.

5.2. Research of Open Issues

5.2.1. Quantification of the Features Offered by the Native Client

In RD 1.2 it was postulated that the Web client of VP should offer all functions of the Native client. If this is not possible or feasible at once, it should be implement in an incremental approach or offer only functions which are demanded by the customers.

The task resulting from this requirement is to prioritise the features specified by the requirements specification [Seh05] according to the significance which each feature has to the customer.

The significance was assessed with the help of the questionnaire "Features of a Web-based Interface to VP" as described in section 2.4.2.1. Succeeding, a table with the results is shown, listing the general significance of each feature, that is the mean of the assessed values expressed in degree of significance. The original structure of the questionnaire has been changed to be conform to the module description given in 3.2[1]. On account of the same reason the description of features and the used terminology have been modified in some cases.

[1] As already stated in section 3.2.3, a 100 percent clear distinction between the modules cannot be made. As the module partition at VP-GE is slightly different from the one chosen in this thesis, the questionnaire had to be structured differently.

Table 5.1.: Quantification of features which are offered by the Native client and shall be offered by the Web client.

Feature	**Significance**
Patient Management Module:	
Ability to search for patient records	High
Ability to manage (create, alter, delete) patient records	Medium
Ability to view orders	Medium
Ability to manage (create, alter, delete) orders	Low
Patient Scheduling Module:	
Ability to view appointments	High
Ability to manage appointments	Medium
Archive Module:	
Ability to view the change history of a patient record and all its data	Medium
Examination Module - In General:	
Ability to view VP-Report	High
Ability to view previously printed VP-Reports	Medium
Ability to view Findings Masks	Medium
Ability to print VP-Report for assigning doctor	High
Ability to alter the VP-Report	Medium
Ability to alter content of the Findings Masks	Medium
Assign or unassign an order	Medium
Assign or unassign a DICOM study	Medium
Export a case to Trium CTG Online	Medium
Display Charts	High
Measurement data transfer from Ultrasound devices	Low
Examination Module - Image Gallery Submodule:	
Ability to view single frame images	High
Ability to view multi frame images	High
Ability to review raw data (Voluson 4D View or LOGIQworks FX)	Medium
Perform measurements on images	Medium
Import image data data via DICOM interface	Low
Digitise image data with the help of a Framegrabber	Low
Digitise image data with the help of a `TWAIN` compatible scanner device	Low
Import image data from file system	Low
Import image data from clipboard	Low
Export image data via DICOM interface	Low
Export image data to file system	Medium
Examination Module - Printing and Faxing Submodule:	
Ability to print	High
Ability to fax	Medium
Examination Module - Rule-Checking Submodule:	
Ability to check plausibility of findings	Low

Comment: The importance of the feature "Ability to check plausibility of findings" was calculated from four instead of five votes as this feature is not available in the USA. Thus, the vote of WG was not counted.

In general no feature has been valuated with "not needed", meaning that at least the final version of the Web client should offer all features of the Native client. This is consistent with the answers given to the question of whether the Web client should be a replacement of and no alternative to the Native client. Here, the majority of the respondents (three out of five) see the Web client as a replacement for the Native client, although they valuated this feature with low significance.

There is one crucial limitation to the requirement that the Web client should offer nearly all functions of the Native client and could be a complete replacement of it. The Web client will never be able to control and access hardware adequately as demanded by RD 8.2 (also see WP 32 for details). This can be reasoned by the following consideration. In order to control and access hardware adequately, the Web client needs to make extensive use of Web browser extensions. If available at all, those Web browser extensions need to use OS functions to be able to access and control hardware. Thereby, the Web client has a strong dependence on the supported Platforms and Web browsers. Moreover only a limited number of Platforms and Web browsers could be adequately supported. Apart from that, there are many other disadvantages, like considerable installation and configuration efforts on each workstation. In other words, the control and access of hardware by the Web client is contradictory to the original idea of a Web client and should not be supported. Consequently, the Web client should be an alternative UI to but no replacement for the Native client.

Still, printers are indirectly supported by the Web client. As every Web browser offers means to print out Web pages, the Web client can indirectly interface with printers, although this is not one of its functions.

Nevertheless, the quantification of the features should be further discussed before starting with the actual implementation. Especially on account of that the demand for a certain feature is only one aspect. The contrary aspect are the connected implementation efforts and constraints set by other requirements which, in the worst case, may completely change the priority.

5.2.2. Identification of Application Components where VP is Needed

In RD 1.3 it was postulated that the Web client should provide access to VP from all application components where access to VP is required.

The task resulting from this requirement is to identify all application components where access to VP is actually needed.

As already mentioned in the introduction, there are several application components where access to VP is needed. The following list gives a detailed overview of these application components, also including the ones where up until now data of VP is provided on a paper-based basis. In this context, paper-based means that the VP-Report is printed out and transfered to a person or institution by mail. Figures 3.13 and 4.2 illustrate this issue.

- *In the functional diagnostic unit:*
  - In **examination rooms** for documentation.
  - At **receptions** for patient identification and examination planning.
  - At **doctor's offices** or **nurses' stations** for completing or reviewing the finding.
  - In **conference rooms** for reviewing daily examinations.
- *In the hospital:*
  - In **other functional diagnostic units** that patients are assigned to for further diagnostics or have been assigned from.
  - At **wards** where inpatients are accommodated and require adequate care and therapy.
  - At **lecture rooms** for reviewing findings for educational purposes.
- *Outside the hospital:*
  - At **other branches** of the hospital, for example those located at another site, which can comprise other functional diagnostic units or wards.
  - At **other healthcare institutions** that the patients are assigned to for further treatment or have been assigned from, for example the **general practitioner's office**.
  - At **specialised healthcare institutions** to gather a second opinion and assistance.
  - At the medicating doctor's or staff's **home** to complete or review the findings.

The sites listed here can be classified into two groups: Local sites (in the hospital) and remote sites (outside the hospital).

5.2.3. Identification of Platforms and Browsers which Should Be Supported by the Web Client

In RD 2.1 and RD 2.2 it was postulated that the Web client should support as few Platforms, OS versions and Web browsers as are needed to allow access from all application components where access to VP is required.

The task resulting from this requirement is to identify all Platforms, OS versions and Web browsers which are used at application components where access to VP is needed.

Application components where access to VP is needed have been identified in section 5.2.2. It is clear that for accessing VP a computer is required at the local or remote site. Normally, most of the sites are already equipped with computers which are required to run other application components like PMS or CDMS.

The questions at hand are which type of Platform is each computer, and which OS version and Web browser is installed.

First, it was tried to find convenient statistics about the market shares of Platforms, OS versions and Web browsers used in the healthcare sector. Several sources were consulted

like the Federal Statistical Office Germany, the German Institute of Medical Documentation and Information, PubMed and the IEEE Computer Society Digital Library, but nothing useful was found.

In the second attempt, it was tried to find out what market shares the Platforms, OS versions and Web browsers have in general by searching the Internet. Here, statistics have been found, although they varied greatly. As example, statistics taken from `www.TheCounter.com`[2] [The06] and statistics taken from `www.W3Schools.com`[3] [W3S06] are compared in table 5.2. The figures are only shown for OS versions and Web browsers which were listed in both statistics, the observation period is March 2006. The figures of [W3S06] have been rounded to integers for a better comparability.

Table 5.2.: Comparison of statistics taken from www.TheCounter.com [The06] and www.W3Schools.com [W3S06] in march 2006.

	[The06]	**[W3S06]**
Web browsers		
Microsoft Internet Explorer 6	88%	59%
Firefox	7%	25%
Microsoft Internet Explorer 5	2%	5%
Opera	1%	2%
OSs		
Microsoft Windows XP	82%	73%
Microsoft Windows 2000	7%	12%
Microsoft Windows 98	5%	2%
Apple Macintosh	2%	4%
Microsoft Windows NT	0%	0%
Linux	0%	3%
Sample size (# of visitors):	227 Mio.	5 Mio.

One may now start to wonder what reasons there are for such big differences. As this would be off the scope of this thesis, it will not be discussed further. For more information please refer to [Ups06], for example. Apart from an overview of different Web browser statistics this Web site also offers an analysis of bias sources.

On account of the significant differences in the cited statistics, it was decided to take them just as guideline. Both statistics suggest that Microsoft still has the majority of market shares of OSs and Web browsers, while Firefox in the field of Web browsers also holds a considerable amount of market shares. In order to figure out if support of other Platforms is also required, their values of significance were acquired with the help of the questionnaire "Features of a Web-based Interface to VP" as described in section 2.4.2.1. Furthermore, the significance to also support Firefox, the second most spread Web browser on the market, in contrast of supporting Microsoft Internet Explorer was assessed.

[2] *Note:* www.TheCounter.com is a commercial Web site offering so-called counters which can be installed on Web sites and are used to gain information about the numbers and characteristics of Web page visitors. The statistics shown here are based on data assessed from all of their counters on the Internet.

[3] *Note:* www.W3Schools.com is a Web site offering freely available tutorials about Web technologies. Statistics shown here are based on data assessed from visitors of their Web site.

The following significance as shown in table 5.3 has been assessed.

Table 5.3.: Significance of supporting referring OS and Web browser.

Feature	**Significance**
Support Web browsers running on Linux	Medium
Support Web browsers running on Apple Macintosh	Low
Support Web browsers running on Sun Solaris	Not needed
Support Firefox	Medium
Support Microsoft Internet Explorer	Medium

Comment: Firefox and Microsoft Internet Explorer have been valuated by the respondents with equal significance. This implies that both Web browsers should be supported by the Web client and not only one of both.

Based on these results and RD 2.1 and RD 2.2 it was decided to support the following OSs and Web browsers as shown in table 5.4 on the Intel *x86* hardware architecture.

Table 5.4.: Web browser and OSs which should be supported by the Web client.

Web browser	**OS**
Microsoft Internet Explorer	Microsoft Windows XP Microsoft Windows 2000
Firefox	Microsoft Windows XP Microsoft Windows 2000 Linux

No specific Service Pack version and no specific Web browser version was defined to achieve a maximum of compatibility. Furthermore, Microsoft Internet Explorer 7 is about to be released. It is yet unknown how fast it will spread. If a restriction to certain Service Packs or Web browser versions occurs due to implementation issues, they have to be specified by the implementor.

5.2.4. Identification of Required Web Browser Extensions

In RD 2.3 it was postulated that the Web client shall utilise as few Plug-ins or Helper Applications as are needed to fulfil RD 1.2. In addition, the utilised Plug-ins or Helper Applications have to support the Platforms, OS versions and Web browsers defined by RD 2.1 and RD 2.2.

The task resulting from this requirement is to identify Web browser extensions which are required to implement certain features (as defined by RS 1.2) as a Web browser does not offer sufficient functions to implement these features. Furthermore, the selected Web browser extensions, according to RS 2.1 and RS 2.2, have to support Microsoft Internet Explorer on Microsoft Windows and Firefox on Microsoft Windows and Linux.

In this section only features of the Web client which require the use of Web browser extensions will be identified and possible Plug-ins or Helper Applications will be given as

example. No specific Web browser extensions will be selected, as this would go beyond the scope of this thesis and should be left to the implementor.

Features requiring no extensions The functions of the patient management module (see section 3.2.4 for details), the patient scheduling module (3.2.5), the statistics module (3.2.6), the case and examination management submodule (3.2.8.1), the findings system (3.2.8.2) and the rule-checking submodule (3.2.8.5) can be realised in Firefox or Microsoft Internet Explorer without requiring any Web browser extensions. This is due to their utilisation of basic UI components like buttons, fields and lists which can be realised by standard Web pages and Web browser functions.

The DICOM interface, the faxing functionality and interfaces to PMS, CDMS and similar application components do also not require any Web browser extensions. This is because they are implemented on the server side and the server itself is the communication partner to the "outside world".

The control and access of hardware by the Web client and the related features and interfaces have already been excluded from the required functions of the Web client in section 5.2.1 and will not be discussed further.

Features requiring extensions The core function of the image gallery submodule (see section 3.2.8.3 for details) is to view image data. As the functions of the image gallery submodule should also be offered via the Web client, it should, if possible, also be realised without using Web browser extensions. In this context, the first approach which might come up into mind could be to convert the image data from DICOM, the storage format of VP (for details see section 3.3.1.2) to standard Internet media formats like `JPEG`. `JPEG` pictures and other Internet media formats can be displayed by all Web browsers without requiring any extensions.

However, by converting DICOM image data to common Internet media formats, additional information stored in DICOM files, for example patient demographics, scale, overlay etc. would not be transfered and therefore be omitted. Furthermore, in Internet media formats a higher compression as common to DICOM files is aspired. This would lower the quality of the image data and thereby contradict with RS 7.5.

Thus, the preferred method should be to display DICOM image data directly through the Web client. As DICOM is no common Internet media format and is, thus, not supported by Web browsers in general, the usage of Web browser extensions is required. A sample of extensions offering the Web-based view of DICOM image data is given subsequently. They are all Java Applets offering DICOM viewing capabilities. As a Java Applets runs as a separate process, it can be classified as Helper Application (see section 2.6).

DICOM Viewer: DICOM Viewer is Open Source and developed by the Nagoya Institute of Technology, Iwata laboratory, Japan. It can be found at `http://mars.elcom.nitech.ac.jp/dicom/index-e.html`. It was, for example, used by Azevedo-Marques et al. in their approach of Web-based RIS-PACS Integration at the University Hospital of Ribeirão Preto [AMCBS04].

EViewBox: EViewBox is Open Source and can be found at `http://eviewbox.sourceforge.net`.

CHILI/Web-Viewer: CHILI/Web-Viewer is commercially available from the Chili GmbH (`http://www.chili-radiology.com`). It was originally developed at the German Cancer Research Centre Heidelberg with the intention to distribute radiological images from PACS to EHR Web-based [MEAM03, EMSM04, ESM+05].

RemoteEye: RemoteEye is commercially available from NeoLogica s.r.l. (`http://eng.neologica.it`). Sample screenshot is shown in figure 5.1.

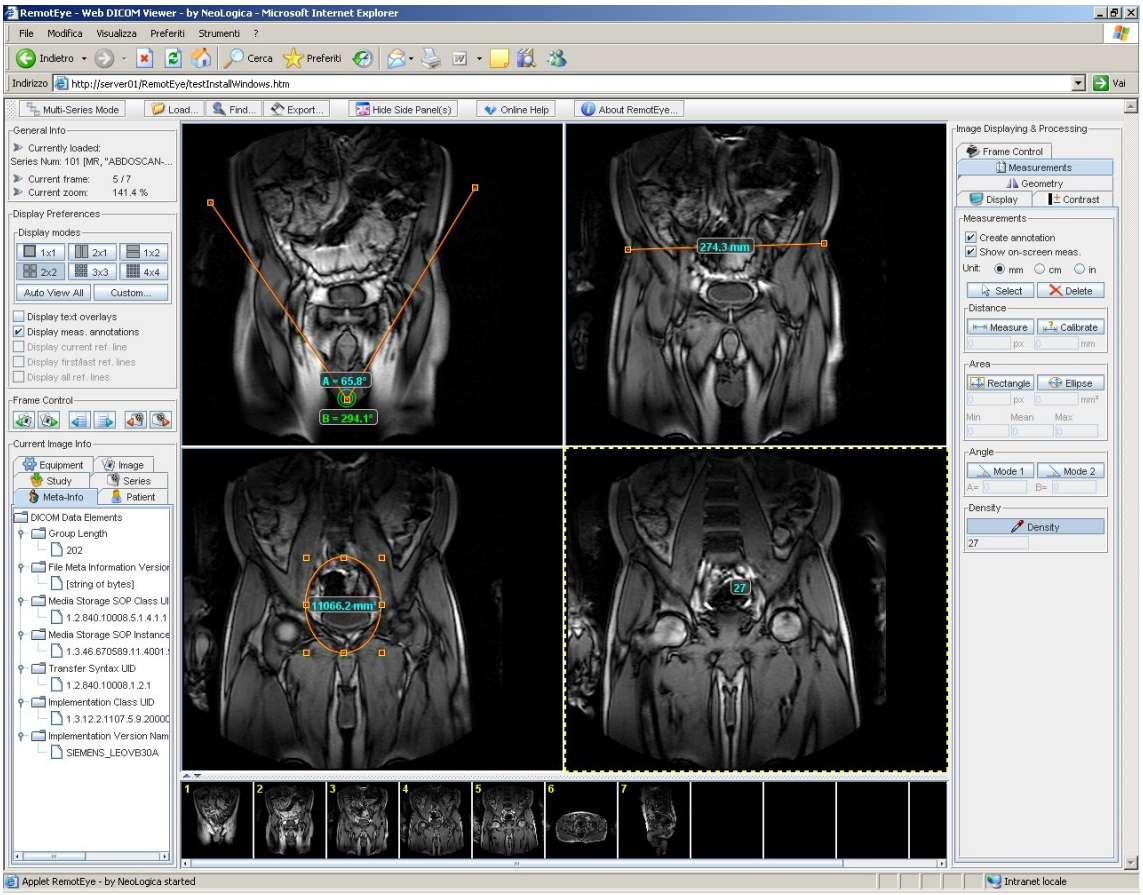


Figure 5.1.: Sample screenshot of Java Applet RemoteEye of NeoLogica s.r.l., taken from [Neo06].

Only Java Applets offering DICOM view capabilities have been listed because other Web browser extensions found did not fulfil RS 2.1 and RS 2.2. A useful overview of Web-based DICOM viewers and other DICOM tools can be found at `http://www.sph.sc.edu/comd/rorden/dicom.html#links` or at `http://dicom.online.fr/fr/dicomlinks.htm`.

LOGIQworks FX and Voluson 4D View which are utilised by the image gallery submodule are also candidates which require Web browser extensions. As stated above, due to their complexity, they will not be discussed in this thesis.

In RD 3.1 it was postulated that the Web client should keep the needed bandwidth as low as possible. In RD 3.3 it was postulated that the Web client should perform and respond in such a manner that the user is not hindered in doing her/his actual work. These two requirements can also be fulfilled by using Web browser extensions to reduce the data

which is exchanged between client and server to react to user interactions.

It should, for instance, be checked, if the data entered in a Web page by the user is in the right syntax. By using standard Web browser functionality, data entered by the user at the client is sent to the server in a so-called commit activity. There it is validated and if the syntax is wrong, the related Web page is sent back to the user containing the original data and some kind of warning message. This behaviour is common for banking or shopping Web sites on the Internet. So for the simple purpose of syntax checking, the data entered by the user is sent to the server and the complete Web page is retransmitted to the client causing actual dispensable network traffic. Moreover if the connection has a low bandwidth or long response time, it will take some time until the refreshed Web page is transfered to the client and displayed to the user.

This handicap can be eliminated by using Web browser Plug-ins. As described in section 2.6, Plug-ins can access and alter the current Web page as they are executed within the Web browser's process. Thereby, means are offered to respond to user interactions without transferring any data between client and server. In reference to the given example, this means that the syntax of the entered data is checked immediately by the Plug-in responsible at the client side. The most popular Plug-in used for these purposes is Java Script.

5.2.5. Interaction with Other Application Components

In RD 6.1, RD 6.2, RD 6.3 and RD 6.3 it was postulated that the Web client should offer means to switch between itself and other application components without having to reenter login information or patient identification information to preserve the context.

The task resulting from this requirements is to identify means to preserve the patient context while switching between different application components and to offer SSO capabilities.

In general, the CCOW CMA as described in section 2.8 is the preferable solution to fulfil RD 6.1, RD 6.2, RD 6.3 and RD 6.3. CCOW was intended right from the start to support context integration in healthcare and, thus, addresses most of the issues in this field. In addition, it is an open and freely available standard.

One disadvantage of CCOW is that it is quite young and not as popular and well-known as HL7, for example. In the questionnaire "Features of Web-based Interface to VP" the respondents valuated the significance of the Web client to support CCOW as low. It was noticeable that the European respondents have not been even aware of CCOW, which actually implies that no customer demanded for it in Europe. This impression was confirmed while completing the product analysis in section 6.2 in which none of the respondents have been familiar with CCOW. Moreover, during literature research no sufficient field reports about CCOW have been found.

Due to the lack of popularity of CCOW one should also consider alternatives while implementing the Web client.

One conceivable alternative to preserve the context when switching from another application to the Web client of VP would be to offer functions for opening a certain patient record via a specific URL. The URL, for example `http://viewpoint.myhospital.de/open?patid=12345`, will open the patient record identified by the patient ID 12345 in the

Web-based UI of VP. This feature is especially useful in case of a Web-based distributed Electronic Patient Record (EPR)[4] where non-redundant data of a certain patient is stored in different application components. The application components form a virtual single EPR by means of integration. The referencing between the components is realised by using unique HTML-links.

A conceivable alternative to offer SSO capabilities would be to utilise functions offered by the OSs like for example Microsoft Enterprise SSO Service.

5.3. Requirement Specification

An overview of the following requirement specification can be found in table 7.1 on page 108.

I. General requirements

RS 1.1: VP shall provide a Web client.
Specification of: RD 1.1
Details: -

RS 1.2: The Web client shall provide all functions of a Native client as specified by the requirements specification [Seh05]. The only functions which are excluded are features which require the control and access of hardware (see WP 32 for details).
Specification of: RD 1.2
Details: In case the Web client is implemented in an incremental approach, features should be added based on their significance given in table 5.1. For every feature the significance should be weighed up against its actual implementation efforts before the decision in which version the feature should be released, is made. See section 5.2.1 for details.

RS 1.3: The Web client shall provide access to VP from local and remote locations.
Specification of: RD 1.3
Details: See section 5.2.2 for details.

II. Requirements concerning supported Platforms, Web browsers and required Web browser extenions

RS 2.1: The Web client shall support Microsoft Windows XP, Microsoft Windows 2000 and Linux on the Intel *x86* hardware architecture.
Specification of: RD 2.1
Details: See section 5.2.3 for details.

RS 2.2: The Web client shall support Firefox on Microsoft Windows and Linux and Microsoft Internet Explorer on Microsoft Windows.
Specification of: RD 2.2
Details: See section 5.2.3 for details.

[4] The Electronic Patient Record is "a complete or partial patient record stored on an electronic storage medium." [HAWB04]

RS 2.3: The Web client shall utilise Web browser extensions to support the display of image data and to fulfil RS 3.1 and RS 3.3. The chosen Web browser extensions have to fulfil RS 2.1, RS 2.2 and RS 3.4.

Specification of: RD 2.3

Details: See 5.2.4 for details.

III. Requirements concerning quality of supported networks

RS 3.1: The Web client shall keep the needed bandwidth as low as possible.

Specification of: RD 3.1

Details: The need of a low bandwidth can be achieved by using Plug-ins. They can carry out certain computation, for example user interaction on the client side, without requiring any client server communication. See section 5.2.4 for details.

RS 3.2: The Web client shall also support unstable networks.

Specification of: RD 3.2

Details: Unstable networks can be supported by offering sufficient means against connection losses and changes of the IP address of the client. For example, if the connection is lost while the user is entering data, after the connection has been reestablished, the user should have the possibility to continue entering data instead of restarting.

RS 3.3: The Web client shall perform and respond in such a manner, that the user is not hindered in doing her/his actual work.

Specification of: RD 3.3

Details: This requirement is in large part dependent on RS 3.1 and can be achieved by using Plug-ins as described in section 5.2.4.

RS 3.4: The Web client shall utilise only standard ports like `HTTP` port 80 or `HTTPS` port 443.

Specification of: RD 3.4

Details: This requirement is in large part dependent on RS 2.3

IV. Requirements concerning data security

Comment: The requirements concerning data security cannot be specified in more detail without knowing the actual technology used. The selection of that technology should be left to the implementor. Furthermore, analysis and comparison of theses technologies would go beyond the scope of this thesis. Still, some advice should be given.

RS 4.1: The Web client shall assure secrecy and confidentiality by offering sufficient means for authorisation.

Specification of: RD 4.1

Details: Potentials and constraints of SSO technology should also be carefully considered while implementing this requirement. CCOW also enables deploying of highly secure SSO solutions.

RS 4.2: The Web client shall offer sufficient means for authentication.
Specification of: RD 4.2
Details: Potentials and constraints of the Health Professional Card (HPC) [5] should also be carefully considered while implementing this requirement, although the respondents of the questionnaire "Features of Web-based interface to VP" valuated that supporting the HPC has only low significance.

RS 4.3: The Web client shall offer sufficient means for non-repudiation and accountability.
Specification of: RD 4.3
Details: -

RS 4.4: The Web client shall offer sufficient means for integrity control.
Specification of: RD 4.4
Details: -

RS 4.5: The technology chosen to address RS 4.1, RS 4.2, RS 4.3 and RS 4.4 shall fulfil all legal regulations of the countries where the Web client is to be employed.
Specification of: - *(new)*
Details: A detailed specification of the legal regulations would go beyond the scope of this thesis and should be done by the implementor. In Germany, these are, for example, Bundesdatenschutzgesetz (BDSG), Informations- und Kommunikationsdienste-Gesetz (IuKDG), Teledienstegesetz (TDG), Teledienstedatenschutzgesetz (TDDSG), Telekommunikationsgesetz (TKG), Signaturgesetz (SiG) etc.. An example for the USA would be HIPAA.

RS 4.6: The Web client shall offer means to protect against data loss.
Specification of: RD 4.5
Details: Data loss in case of connection loss during data entry is addressed by RS 3.2. Data may also be lost if the Web browser quits unexpectedly or the workstations locks up unexpectedly during data entry. Here, the data already entered could, for example, automatically be sent to the server at regular intervals to prevent the loss of all data in case of failure. Still, it is clear that no full protection against data loss can be guaranteed by a Web client.

V. Requirements concerning installation and maintenance

RS 5.1: The Web client shall require no or at least a minimum of installation and configuration efforts.
Specification of: RD 5.1
Details: In reference to section 5.2.4 at least some Web browser extensions are required. They are linked to installation efforts. In order to enable its

[5] The Health Professional Card is an individually programmed access card for health care professionals. The HPC is used to assign read and/or write access to specific data fields on the patient card. The chip of the card stores the doctor's ID, read and/or write access rights and a digital signature (for secure data transmission) [Gie06]. There are country specific standardisation attempts. For more information on, for example, the German approach, please refer to `http://www.bundesaerztekammer.de/30/eArztausweis`.

usage, changes at the default configuration of the Web browser will also be inevitable. Still, these efforts should be kept to a minimum.

RS 5.2: The Web client shall require no or at least a minimum of maintenance efforts.

Specification of: RD 5.2

Details: In reference to 5.2.4 at least some Web browser extensions which are linked to maintenance efforts are required. Still, these efforts should be kept to a minimum.

VI. Requirements concerning interaction with other application components

RS 6.1: The Web client shall offer means to switch from itself to another application component running on the same workstation without having to reenter login information.

Specification of: RD 6.1

Details: CCOW should be the preferred standard to address this requirement. Other technologies like SSO should also be considered. This requirement is also related to RS 4.1. See section 5.2.5 for details.

RS 6.2: The Web client shall offer means to switch from another application component running on the same workstation to itself without having to reenter login information.

Specification of: RD 6.2

Details: See RS 6.1.

RS 6.3: The Web client shall offer means to switch from itself to another application component running on the same workstation without having to reenter patient identification information to preserve the patient context.

Specification of: RD 6.3

Details: CCOW should be the preferred standard to address this requirement. Alternatively the patient context can also be passed via specific URLs. See section 5.2.5 for details.

RS 6.4: The Web client shall offer means to switch from another application component running on the same workstation to itself without having to reenter patient identification information to preserve the patient context.

Specification of: RD 6.4

Details: See RS 6.3.

VII. Requirements concerning the UI

RS 7.1: The UI of the Web client shall be based on the existing masks definition (see section 3.2.8.2 for details) if possible and feasible. If not possible and feasible, the Web client shall utilise a standardised UI definition language so that the UI of the Native client and the UI of the Web client can be created from a single definition.

Specification of: RD 7.1

Details: A detailed analysis of available UI definition languages would go beyond the scope of this thesis. An overview of, for example, XML Markup languages for definition of UIs can be found at http://xml.coverpages.org/userInterfaceXML.html. Representatives of this class of user interface definition languages are, for example, the Extensible User Interface Language (XUL) of the Mozilla Foundation or Extensible Application Markup Language (XAML) of Microsoft. There certainly are other classes of UI definition languages but as XML is the most widespread and best-known markup language, it should be the preferred one.

RS 7.2: The Web client shall offer a UI which resembles the one of the Native client in appearance and is controlled similarly.

Specification of: RD 7.2

Details: The "look and feel" should be as close to the Native client as possible.

RS 7.3: The UI of the Web client shall offer more similarity to the UI of Trium CTG Online than the UI of the Native client and should thereby point out the close association between both products.

Specification of: RD 7.3

Details: Trium CTG Online will be analysed in section 6.2.3. In general, it is more desirable to demand Trium Analysis Online GmbH to modify their product's layout and handling so that it becomes similar to VP. The modification of Trium CTG Online would cause much less effort and would not contradict RS 7.2.

RS 7.4: The Web client shall offer a UI which supports automatic testing.

Specification of: RD 7.4

Details: The tools for automatic testing used at VP-GE should be adequately supported by the Web client.

RS 7.5: The Web client shall preserve the quality and integrity of all image data provided to the user through the Web-based UI.

Specification of: RD 7.5

Details: This requirement has been addressed in section 5.2.4. It must be carefully determined that the chosen Web browser extension(s) for displaying the image data fulfil(s) this requirement.

VIII. Requirements concerning hardware

RS 8.1: The Web client shall keep its hardware requirements to a minimum.

Specification of: RD 8.1

Details: The hardware requirements of the Web client result from the hardware requirements of the Web browsers and Web browser extensions used.

RS 8.2: The Web client shall offer no means to control hardware devices connectable to VP.

Specification of: RD 8.2

Details: The original definition of this requirement (RD 8.2) has been disproved in section 5.2.1. The Web browser itself offers features to connect to printers which can also be utilised by the Web client.

6. Technology Comparison

6.1. Outline

In the preceding chapter the requirements of a Web client have been specified in detail.

The aim of this chapter is to analyse and compare certain products which offer a Web-based UI. Furthermore, the Terminal client-server architecture should be considered as an alternative technology to the Web client. In sections 6.2.1 to 6.2.3, the products will be described and analysed. In section 6.2.5, the products will be compared to each other based on the requirements specification set up in section 5.3. Section 6.2.6 will try to evaluate the requirements specification based on the information gathered by the product analysis and comparison. Section 6.3.1 will describe which means are offered by the Terminal client-server architecture to fulfil the requirements specification. Section 6.3.2 will examine to which extent the Terminal server solution fulfils the requirement specification in comparison to the Web client.

The succeeding chapter will draw the conclusion of this thesis.

6.2. Product Analysis and Comparison

As described in section 2.5 product analysis was carried out to determine how other manufacturers realised the requirement specification defined in section 5.3.

The product analysis and comparison is meant to provide the implementor with ideas of how the Web client could be realised.

Details on why especially these products have been chosen can be found in section 2.5.

6.2.1. Euroking – E3

The information given here has been acquired with the help of the questionnaire "Product Analysis" (see section 2.4.2.2 for details). Further information was taken from the Web page of Euroking Miracle Ltd..

6.2.1.1. General Description

Euroking Miracle Ltd. is a partner of GE Healthcare. Apart from distributing VP they also sell their own product E3 in the UK and Ireland.

E3 is a Maternity Information System and belongs to the class of MDS and Patient Data Management System (PDMS). It offers a wide range of functions to optimally support the work flow at an obstetrical department.

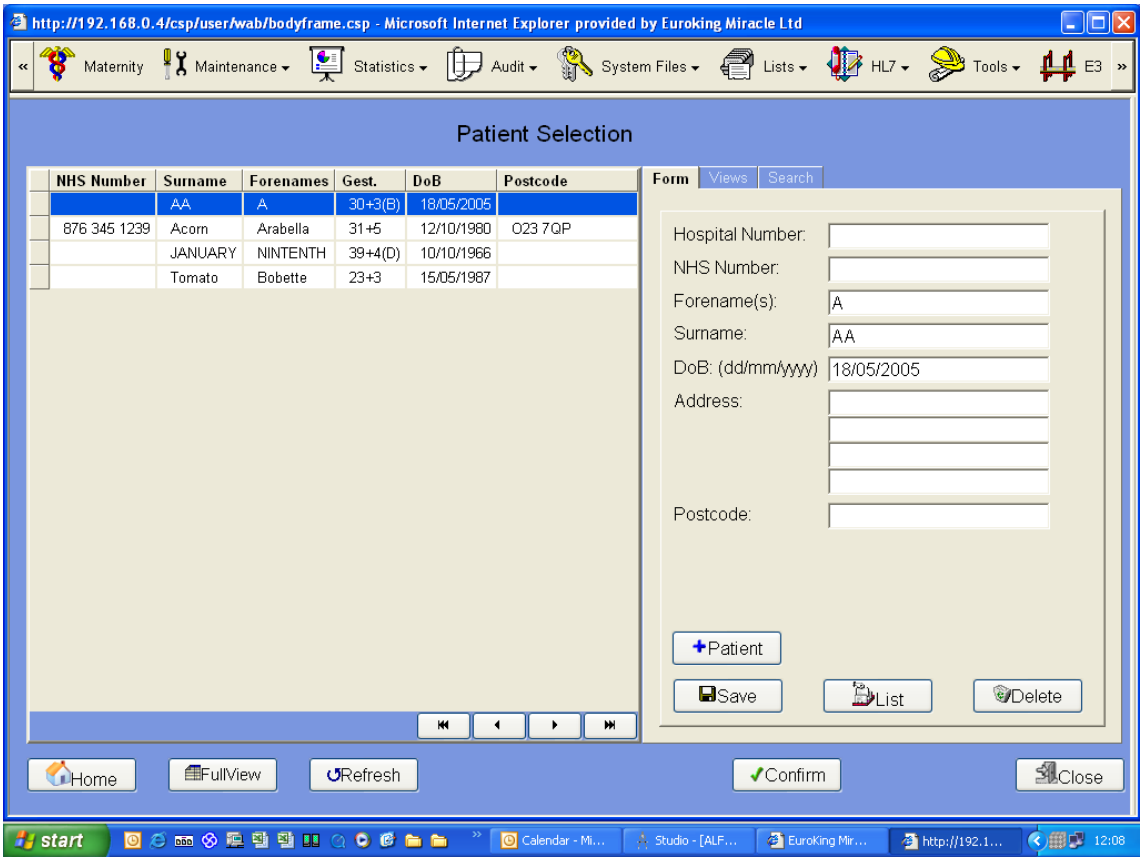


Figure 6.1.: Screenshot showing the patient selection screen of E3.

E3 stores data in a patient-oriented way. It supports the management of care pathways, risk management, acquisition and storage of partograms, real-time data entry. In addition it can utilise Microsoft Word for report generation. One core feature of E3 is a document management system with which any type of data files like Microsoft Word or Adobe Acrobat Reader documents can be managed. The creation of statistical reports is also supported.

E3 utilises the HL7 standard, `ODBC` and `XML` to interface with other application components. It does not support the DICOM standard, although DICOM files can be handled by the document management system of E3.

6.2.1.2. Technical Aspects

As the National Health Service (NHS) has a special contract with Microsoft, it could be assumed that all potential customers run only Microsoft technology. So Microsoft Internet Information Service (IIS) was chosen as Web server to build on. Microsoft Internet Explorer is the only Web browser and Microsoft Windows is the only OS supported by the Web client. The Java Virtual Machine (JVM) needs to be installed on the workstation to be able to use the Web client. The default configuration of Microsoft Internet Explorer has to be modified so that the execution of Java Script is allowed. Applications to view or edit the file types managed by E3 have to be installed on the workstation which runs the Web client.

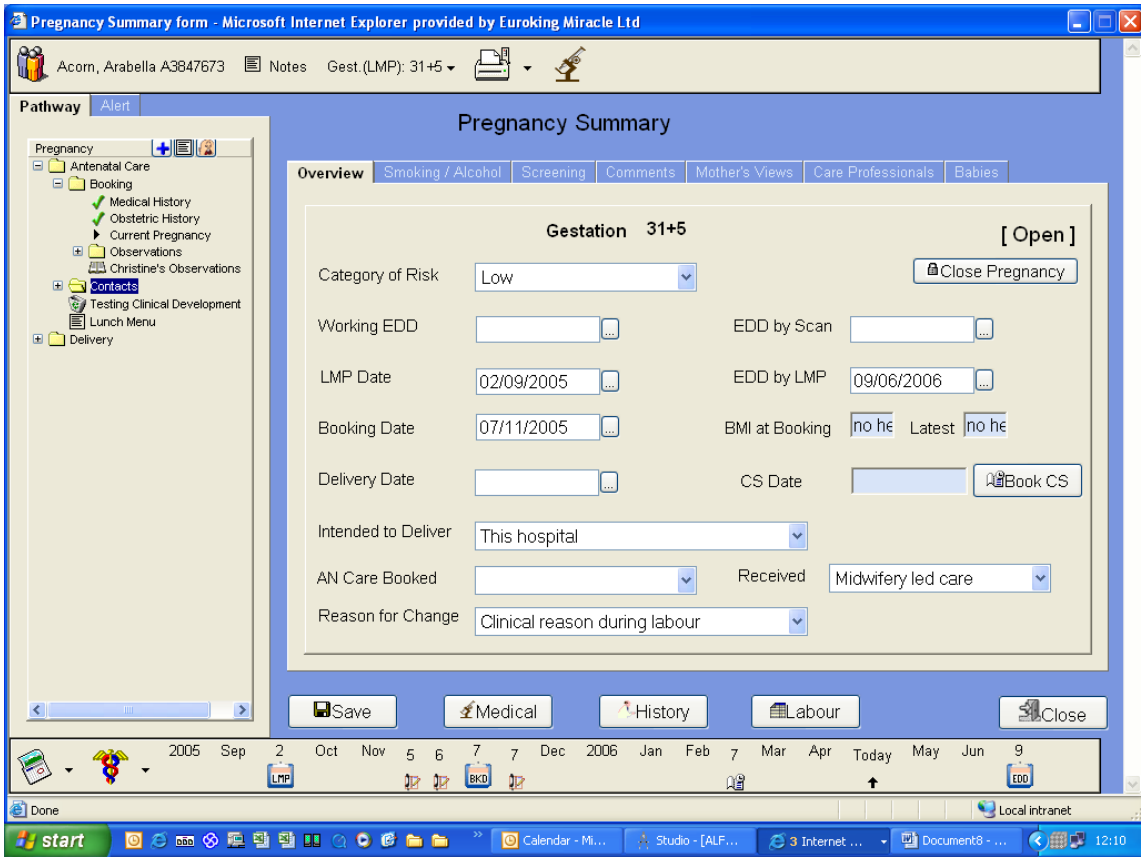


Figure 6.2.: Screenshot showing the pregnancy summary of E3.

In case that updates or security patches are issued for the Web server (IIS), it is the responsibility of the customer to apply them.

For the E3 Web client it is recommended to have a network connection to the server with a bandwidth of at least 100 Mbit/s. Some hospitals run E3 on a 10 Mbit/s LAN in a satisfying quality. Unstable networks are not supported. An unexpected connection loss is a major problem for E3. Still, they offer means to prevent the complete loss of all data entered in a session. One of it is a periodical auto-saving function by which the entered data is saved on the server after a fixed period of time. Another one is that after the completion of a mask or a set of entry fields the entered data is also committed to the server. At the moment they have no capabilities to work offline [1].

The security policy of E3 is based on the following assumption. All users of the LAN are authorised. It is the hospital's responsibility to prevent unauthorised access to the network. So there is no potential danger that packets are viciously modified and integrity control is required.

In future an "Integrated Login" (SSO) will be supported for authorisation purposes. This "Integrated Login" is Microsoft Windows based and is supposed to be introduced NHS-wide. Authentication is performed by Microsoft Windows. All other application components

[1] In a former version of E3 a Native client was also offered with which offline work was possible. It was made for midwives who could synchronise data after their home visits.

retrieve the username from Microsoft Windows. In addition to the "Integrated Login" the HPC could be used, although from the point of view of Euroking Miracle Ltd. it looks that there will not evolve a common standard for the HPC in the UK in the near future.

CCOW is known to Euroking Miracle Ltd. but they doubt that it will become so popular that they will also have to support it. Their doubts are mainly caused by the assumption that third party providers will not agree to allow other applications to control their products. For SSO functionality they rely on the "Integrated Login" as describe above. For providing patient-context based access to E3, they will offer a URL-based solution in the near future. By specifying a certain URL containing the patient identifier, other application components can open the related patient record directly in E3.

If Euroking Miracle Ltd. had the chance to redesign E3 from scratch, they would use another DBMS. They are content with their current DBMS Intersystems Cache but would choose a different one on account of popularity reasons. Furthermore, they would utilise .NET to increase the performance and offer better means for integration. Additionally, they would focus on offering access to E3 from anywhere.

6.2.2. GE Vingmed Ultrasound – WebEx

The information given here has been acquired with the help of the questionnaire "Product Analysis Questionnaire" (see section 2.4.2.2 for details). Further information was taken from the Web page of the GE Vingmed Ultrasound.

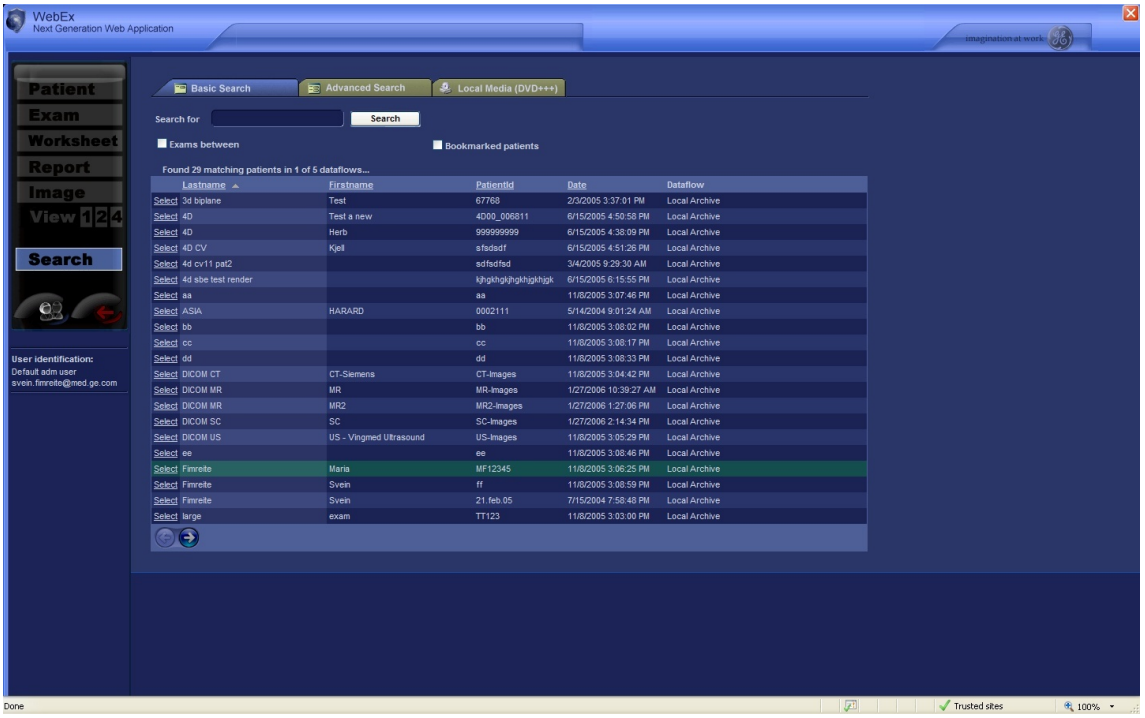


Figure 6.3.: Screenshot showing the search screen of WebEx.

6.2.2.1. General Description

GE Vingmed Ultrasound is a GE Healthcare company and the manufacturer of EchoPAC. EchoPAC is sold worldwide.

EchoPAC is, as the name already suggests, a PACS for image data originating from US devices. It stores data in a patient-oriented way. Its functional range includes classical PACS functionality as well as features for advanced analysis of image data and report generation. Especially raw image data acquired by Vivid US devices is extensively supported by these advanced analysis tools. EchoPAC is marketed as part of the Vivid product family. As Vivid scanners are popular in cardiology, EchoPAC's main area of application are cardiological departments at hospitals.

EchoPAC interfaces with PMS, CDMS and similar application components by means of the HL7 standard.

WebEx is supposed to be the future Web client to EchoPAC, offering similar functions. It is reusing shared components of EchoPAC. Until now it was developed by S. Fimreite in his spare time and has not yet reached official status, meaning that it is not yet sold to the customer and still in the evaluation and testing phase. Specialities of WebEx are that viewing on mobile devices is also supported and that it offers full-screen view capabilities and can be displayed on systems with two screens attached. Moreover WebEx offers predefined texts for report generation and the performance of measurements on image data. It aims at supporting any type of DICOM imaging modalities in a hospital.

WebEx can in general connect to any type of database by means of ODBC. At the moment it only supports the database structure of EchoPAC.

Figure 6.4.: Screenshot showing the image gallery screen of WebEx.

6.2.2.2. Technical Aspects

IIS was chosen as Web server due to GE Vingmed Ultrasound's choice of Microsoft Visual Studio .NET as Integrated Development Environment (IDE). Therefore, a Microsoft product was also chosen as Web server. The Web-based UI is conform to the Extensible HyperText Markup Language (XHTML) standard. In general, every common Platform and Web browser should be supported. WebEx was successfully tested with Microsoft Internet Explorer, Netscape and Opera running on Microsoft Windows, Linux and Apple Macintosh. The default configuration of the Web browser has to be modified to allow Pop-Ups. In addition, a media player has to be installed which is capable of viewing Microsoft Windows Media, MPEG-1 or MPEG-4. The player is not required to have an interface to the Web browser as videos are downloaded onto the workstation and can than be opened from the file system. Additionally, a ZIP compatible packing programme needs to be installed to be able to unpack downloaded videos.

In case that updates or security patches are issued for the Web server (IIS), the Microsoft Windows Automatic Update functionality is used to keep the server up-to-date.

WebEx has two ways for handling image data. If image data is only to be reviewed, the WebEx server converts it to standard Internet media formats like JPEG or MPEG-4 and shows it embedded in the related Web page. If it is, for example, to be post-processed in real-time by EchoPAC, it is compressed in an ZIP file on the server and can be downloaded to the workstation.

Concerning the networks supported by WebEx and their quality, WebEx actually requires no minimum bandwidth. Still, constraints on the required bandwidth are set by the time which is seen as acceptable to load the image data from the server. For example loading a full screen image loop over an Integrated Services Digital Network (ISDN) line would take around 2 minutes[2]. Thus, completely loading an examination record would take at least eleven minutes, assuming that usual examination records comprise image data of the size of five or more full screen image loops.

Unstable networks are not supported. If the connection is lost, the default warning message for offline status of the Web browser is displayed to the user. If the connection loss occurred during a download process, the download can be resumed in case it was only a temporary interruption.

As WebEx is also designed for Internet usage, security is assured by a number of technologies. Authorisation and authentication are supported with the help of usernames and passwords. The HPC may be supported in future, depending on its implementation efforts and the significance it has to the customer. The logged in users are automatically logged off after a certain period of inactivity. Accountability is assured by logging user activity. Integrity control is guaranteed by using the HTTPS protocol with 128 bit encryption. The Web pages of the Web-based UI expire immediately. If they have to be reloaded, they will be directly transfered from the server. So a potential attacker does not have a chance to maliciously modify a temporarily stored copy of a Web site.

Furthermore, integrity of image data downloaded at the client is assured by compressing

[2] A full screen image loop has a size of about 1 MBytes. For example this could be one heart cycle recorded with a frame-rate of 40-60 fps and a resolution of 1024×768 pixels. An ISDN line has a bandwidth of 64 kbit/s. This implies that the corresponding transfer time of one full screen image loop is at least $\frac{1 \times 2^{20} \times 8 \text{ Bit}}{64000 \text{ Bit/s}} \approx 131 \text{ s} \approx 2.2 \text{ min}$

it in a password protected ZIP file.

The use of CCOW or other means for contextual integration have not yet been evaluated for WebEx.

At the moment GE Vingmed Ultrasound is content with the technology chosen for WebEx and would not change anything if they had the chance to reimplement it from scratch.

6.2.3. Trium CTG Online

The information given here has been acquired with the help of the questionnaire "Product Analysis Questionnaire" (see section 2.4.2.2 for details). Further information was taken from the Web page of the Trium Analysis Online GmbH.

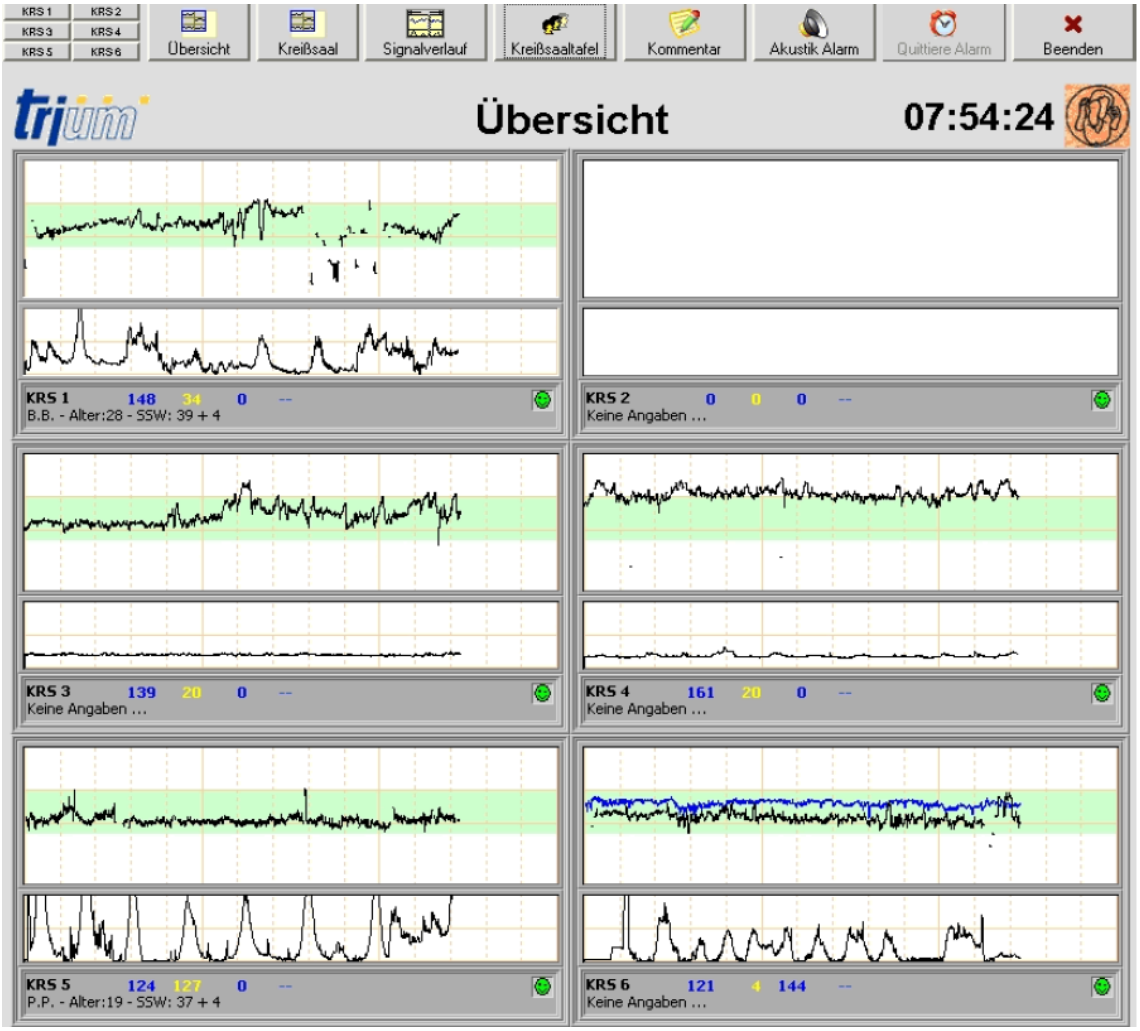


Figure 6.5.: Screenshot showing the overview screen of Trium CTG Online.

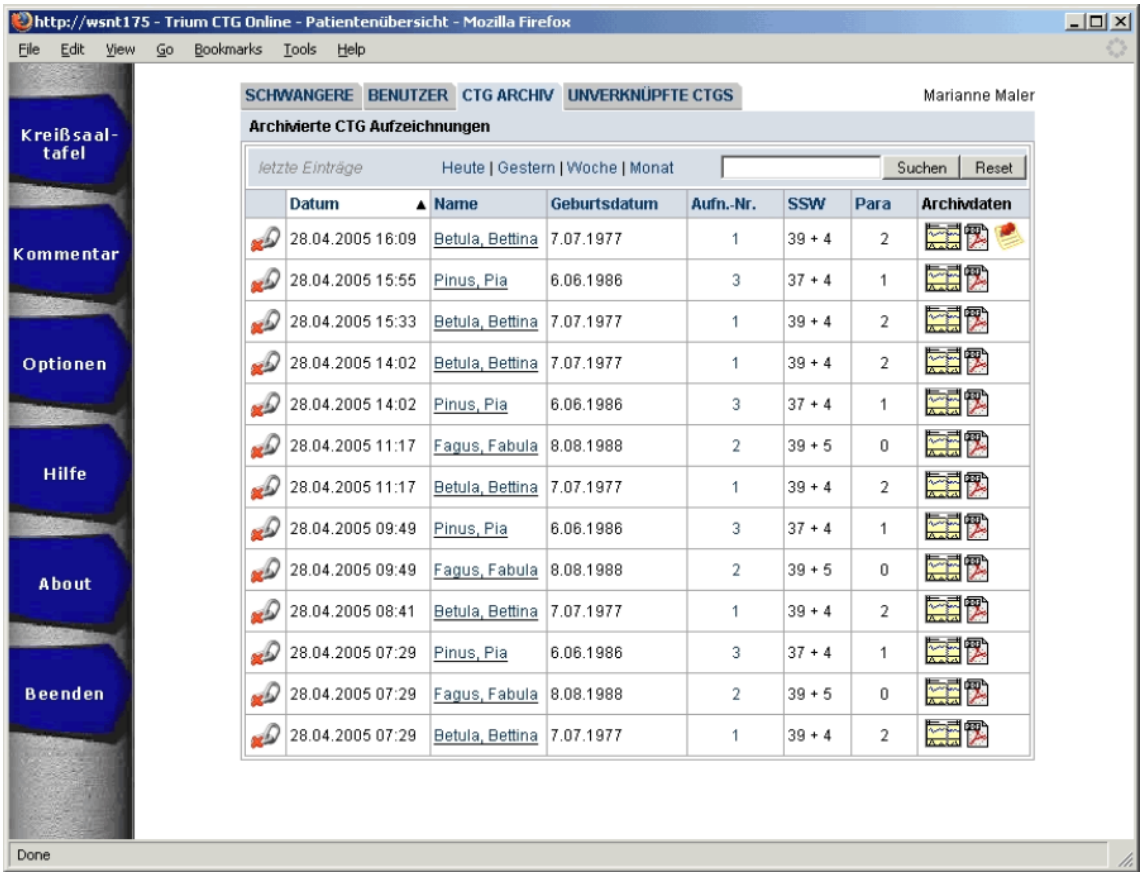
6.2.3.1. General Description

Trium Analysis Online GmbH is a partner of GE Healthcare. They are specialised in providing online Web-based solutions for Telemedicine and Clinical Trials. Trium CTG Online is sold internationally and also marketed by VP-GE as an extension to VP.

Trium CTG Online is a PDMS. It is used for centrally monitoring Cardiotocography (CTG)s at obstetrical departments. All active CTG monitors send their measurement to the Trium CTG Online server where the data is received and recorded. Visualisation is performed by the Web client. It offers some special features like visual or acoustic alarms for abnormal CTGs. Furthermore the Web client can also be utilised via mobile devices, although the common area of application of Trium CTG Online is on workstations connected to the Intranet of a hospital.

The visualisation of CTGs is delivery-room-oriented. The recorded data can be accessed in an patient-oriented way.

Trium CTG Online interfaces with PMS, CDMS and similar application components through the HL7 standard. It has a special interface to VP from which patient demographics and certain details about the current pregnancy are imported.



Datum	Name	Geburtsdatum	Aufn.-Nr.	SSW	Para	Archivdaten
28.04.2005 16:09	Betula, Bettina	7.07.1977	1	39 + 4	2	
28.04.2005 15:55	Pinus, Pia	6.06.1986	3	37 + 4	1	
28.04.2005 15:33	Betula, Bettina	7.07.1977	1	39 + 4	2	
28.04.2005 14:02	Betula, Bettina	7.07.1977	1	39 + 4	2	
28.04.2005 14:02	Pinus, Pia	6.06.1986	3	37 + 4	1	
28.04.2005 11:17	Fagus, Fabula	8.08.1988	2	39 + 5	0	
28.04.2005 11:17	Betula, Bettina	7.07.1977	1	39 + 4	2	
28.04.2005 09:49	Pinus, Pia	6.06.1986	3	37 + 4	1	
28.04.2005 09:49	Fagus, Fabula	8.08.1988	2	39 + 5	0	
28.04.2005 08:41	Betula, Bettina	7.07.1977	1	39 + 4	2	
28.04.2005 07:29	Pinus, Pia	6.06.1986	3	37 + 4	1	
28.04.2005 07:29	Fagus, Fabula	8.08.1988	2	39 + 5	0	
28.04.2005 07:29	Betula, Bettina	7.07.1977	1	39 + 4	2	

Figure 6.6.: Screenshot showing the archive screen of Trium CTG Online.

6.2.3.2. Technical Aspects

Trium Analysis Online GmbH chose Apache as Web server as it is the market leader. It offers several extensions and is from their point of view more secure than most other Web servers. Trium CTG Online utilises the OpenSA server suit which comes with Apache 1.X. In order to provide additional functionality Macromedia ColdFusion is used as application server. The Web-based UI conforms to HTML 4.01. As Web browsers Microsoft Internet Explorer, Mozilla and Firefox running on Microsoft Windows are supported. The JVM needs to be installed on the client workstation and the usage of Java Script has to be allowed by the Web browser. In order to enable acoustic alarms, the installation of Macromedia Flash is required, although usage of Trium CTG Online is possible without it. Moreover, Adobe Acrobat Reader needs to be installed to be able to view archived CTG printings. A JPEG-based viewing of archived CTG printings is also available. Before using the Web client on a workstation for the first time, the SSL root CA certificate has to be installed to enable encrypted data transfer.

In case updates or security patches are issued for the Web server (Apache, OpenSA), it is the responsibility of the customer to install them.

Concerning the networks supported by Trium CTG Online and their quality, Trium CTG Online requires at least a bandwidth of 9600 Bit/s. In the hardware recommendations for Trium CTG Online 100 Mbit/s are defined. Unstable networks are not supported as this would contradict the principle of the online monitoring system which needs to have a constant connection to the server. The user is informed about an established server connection by a rotating foetus in the right upper corner of the Web client. If the connection is lost, the foetus will no longer rotate and a warning message will be displayed to the user.

Security of Trium CTG Online is assured with the help of a set of technologies. Authorisation and authentication is supported with the help of usernames and passwords. The HPC may be supported in future. Accountability is assured by logging user activities. Thus, it is possible to track which users have been logged in at the time an alarm indicating an abnormal CTG occurred. Integrity control is assured by using the HTTPS protocol with 128 bit encryption.

CCOW or other means for context integration will not be supported in the near future.

If Trium Analysis Online GmbH had the chance to reimplement their Web client from scratch, they would use a professional DBMS and build on the latest version of their application server.

6.2.4. Definition of Comparison Criteria

The products introduced in the preceding sections are compared in detail in table 6.1. Comparison criteria have been defined according to the requirement specification set up in section 5.3. They are meant to show how other manufacturers realised certain requirements or issues linked to the requirements. Other criteria covered by the comparison are issues which are or may be relevant for the implementation of the Web client. The information used in this table was assessed with the questionnaire “Product Analysis Questionnaire” as described in section 2.4.2.2.

The used criteria for comparison will be subsequently defined together with the related requirement(s) they aim at.

Criteria which are to give a general overview of the product:

Manufacturer: Name of the company.

Product Name: Name under which the product is sold.

Product Category: Category to which the product belongs to as defined by [HAWB04, 89-100].

Goals: Main goal of the product.

Networks: Network types the Web client was designed for. Aims partly at RS 1.3.

Web page for further information: URL where further information on the product and its manufacturer can be found.

Criteria which are to give an overview of the Web server and its characteristics:

Web server: Type of Web server used to host the Web based access.

Database: Type of DBMS used.

Web server extensions: Web server extensions which are used, like scripting languages etc.

Criteria which are to give an overview of the Web client characteristics and some details on its implementation:

Supported platforms: Aims at RS 2.1.

Supported Web browsers: Aims at RS 2.2.

Required Web browser extensions: Extensions which are required to be installed on the workstation to be able to run the Web client. Also comprises required changes at the default configuration of the Web browser. Aims at RS 2.3, RS 5.1 and RS 5.2.

Distributed media: Media types and data file types which are distributed via the Web client and which may need Web browser extensions to be viewed. Aims at RS 2.3.

HTML version: `HTML` version to which the Web-based UI is conform. Aims at RS 2.2.

Used ports: Network ports utilised by the Web client. Aims at RS 3.4.

Required network bandwidth: The minimum bandwidth required to run the Web client. Aims at RS 3.1.

Support of unstable networks: If and how unstable networks are supported. Also includes means against data loss. Aims at RS 3.2 and RS 4.6.

Authorization: Means used for authorisation purposes. Aims at RS 4.1.

Authentication: Means used for authentication purposes. Aims at RS 4.2.

Accountability: Means used to guarantee non-repudiation and accountability. Aims at RS 4.3.

Integrity Control: Means used for integrity control purposes. Aims at RS 4.4.

Support of CCOW: CCOW or other means supported to guarantee contextual integration. Aims at RS 6.1, RS 6.2, RS 6.3 and RS 6.4.

6.2.5. Product Comparison

In table 6.1 the analysed products are compared by the criteria as defined in section 6.2.4.

Table 6.1.: Comparison of E3, WebEx and Trium CTG Online.

	E3	WebEx	Trium CTG Online
Manufacturer:	Euroking Miracle Ltd.	GE Vingmed Ultrasound	Trium Analysis Online GmbH
Product Name:	E3	WebEx	Trium CTG Online
Product Category:	MDS and PDMS	PACS	PDMS
Goals:	Support the workflow in an obstetrical department with an easy-to-use interface.	High secured Web-based PACS workstation designed to handle any type of imaging modalities.	Web-based distribution and archiving of cardiotocograph readings in real-time.
Networks:	Intranet	Intranet and Internet	Intranet
Web page for further information:	`http://www.euroking.com`	`http://www.gehealthcare.com/usen/cv_ultrasound/products/echo_specs.html`	`http://www.trium.de`
Web server:	IIS	IIS	Apache
Database:	Intersystems Cache	Microsoft SQL Server	Microsoft Access
Web server extensions:	-	Microsoft Active Server Pages .NET (ASP.NET)	OpenSA, Macromedia ColdFusion, G Web server
Supported platforms:	Microsoft Windows	Microsoft Windows, Apple Macintosh, Linux	Microsoft Windows
Supported Web browsers:	Microsoft Internet Explorer	Microsoft Internet Explorer, Netscape, Opera	Microsoft Internet Explorer, Firefox, Mozilla
Required Web browser extensions:	JVM has to be installed and the usage of Java Script has to be allowed. Further applications are required to open files managed by the file management system of E3 (like Microsoft Word, Adobe Acrobat Reader etc.)	Media player which is capable of playing Microsoft Windows Media, `MPEG-1` or `MPEG-4`. `ZIP` tool to unpack downloaded videos. Allow use of Pop-Ups.	Macromedia Flash, Adobe Acrobat Reader. JVM has to be installed and the usage of Java Script has to be allowed. The `SSL` root CA certificate must be installed.
Distributed media:	every data file type	text, images (`JPEG`, `GIF`), videos (`MPEG-4`, `MPEG-1`)	text, images, Macromedia Flash, `Portable Document Format (PDF)`
HTML version:	`HTML` as supported by Microsoft Internet Explorer	`XHTML` 1.0, in some cases `XHTML` 1.1.	`HTML` 4.01
Used ports:	`HTTP` or `HTTPS`.	`HTTP` or `HTTPS`.	`HTTPS` and port 8080

continued on next page

	E3	WebEx	Trium CTG Online
Required network bandwidth:	100 Mbit/s are recommended. There are hospitals operating with 10 Mbit/s.	No minimum bandwidth, depends on how fast image data is to be loaded.	9600 Bit/s, 100 Mbit/s recommended
Support of unstable networks:	Not supported. Offer periodical auto-saving functions to prevent data loss.	-	-
Authorization:	Login with username	Login with username	Login with username
Authentication:	Password	Password	Password
Accountability:	-	Logging of user activities	Logging of user activities
Integrity Control:	SSL	SSL	SSL
Support of CCOW:	-	-	-

6.2.6. Justification of Requirements

In this section the validity and practicability of the requirements specified in section 5.3 will be judged by means of the information gathered during the preceding product analysis and comparison.

A Web client has been a requirement in many tenderings. Thus, it is out of question to offer a Web client for VP as specified by RS 1.1. The major motivation of providing a Web client is to offer access to VP from all places where access is needed as appointed by RS 1.3.

The analysis showed that all products stand out with some special functions, which might at first glance seem difficult to be implemented by a Web-based UI. Still these functions were realised and so it appears to be adequate and realistic to demand that nearly all features of the Native client of VP should be implemented by the Web client as specified by RS 1.2.

Furthermore, the capability of supporting Microsoft Windows and Microsoft Internet Explorer can be found in all products, although every product makes its own choice of the additionally supported Platforms, OS versions and Web browsers. E3 is quite restrictive by only supporting Microsoft technology. They can assume, however, that only due to the special contract of the NHS with Microsoft all of their potential customers utilize Microsoft technology. Thus, they have no need to support any alternative Platforms, OS versions or Web browsers. As VP and its future Web client is intended to be deployed worldwide, it is essential to also support alternative technology as demanded by RS 2.1 and RS 2.2, so that a maximum of possible workstation configurations can be served by the Web client of VP.

Concerning RS 2.3, there is no analysed product which does not require any type of Web browser extensions. It therefore seems adequate to utilise Web browser extensions. Their number should be kept to a minimum, though. All analysed products also require changes of their Web browser's default configuration. This implies that VP's Web client might also need some modifications. Still, it is essential to keep the connected efforts for setup and maintenance of the Web client as low as possible as demanded by RS 5.1 and RS 5.2.

Otherwise the Web client would not offer any outstanding advantages to the Native client.

The applicability of the analysed products is mostly limited to the Intranet of a hospital. Only WebEx explicitly states that it is also designed for Internet usage. Still, the other products can be accessed from remote places as well. The corresponding approach is to offer dial-in facilities for remote users, so that they can connect to the Intranet and thereby have access to the related application component. Here, the responsibility of preventing unauthorised access is in the hands of the hospital and the manufacturer does not need to secure its product for Internet usage.

On the Intranet, all products utilise simple user credentials like username and password for authorisation and authentication purposes. In order to guarantee accountability WebEx and Trium CTG Online are logging the user's activities. The standard technology to assure the integrity of the transmitted data is to use SSL encrypted network connections.

The requirements specified by RS 4.1, RS 4.2, RS 4.3 and RS 4.4 have been acknowledged by the analysed products. The Web client of VP should, however, offer more advanced security features so that it can also be accessed on the Internet and utilise techniques like the HPC or CCOW. As VP is sold worldwide, it has to fulfil all legal regulations like demanded by RS 4.5 when assuring data security.

The access of application components from remote places does not only come along with demands for advanced security. The requirements to support low bandwidth and unstable network connections are also a big issue in this context. Although especially WebEx and Trium CTG Online specify that they need a minimum bandwidth, from a realistic point of view they require at least a bandwidth which is characteristic of LANs (10-100 Mbit/s) to not hinder the user in her/his work. So it is substantial to address the coherence between low bandwidth utilisation and not interfering the user's work in two separate requirements like it was done by RS 3.1 and RS 3.3.

WebEx and E3 see a sudden network interruption like it is common for unstable networks as a major problem while applying the online monitoring tool Trium CTG Online in unstable networks is irrelevant. So it should be considered from the initial implementation of a Web client to offer means to also cope with unstable connections like it was specified by RS 3.2. E3 is the only product which offers means against data loss by the auto-saving functionality. So the fulfilment of RS 4.6 is also essential when implementing a Web client, although a 100 percent prevention of data loss will not be possible.

Concerning the utilised network ports: Apart from Trium CTG Online all products only communicate via standard ports which are most probably forwarded by the hospital's firewall. This is crucial to be also adopted by the Web client of VP as specified in RS 3.4.

From the current point of view, no analysed product does support CCOW or will do so in future. Before implementing CCOW it should be carefully determined which benefits can be taken from it. Especially the question what advantages supporting CCOW have for VP when no other product can interface with it. Perhaps it is sufficient to support the "integrated login" as describe for E3 and to offer some kind of URL based access to specific patient records. In any case, certain means for contextual integration must be provided by the Web client as demanded by RS 6.1, RS 6.2, RS 6.3 and RS 6.4 to strongly facilitate the health care professionals in doing their actual work.

The requirement RS 7.1 has not been addressed by the product comparison. Still, the idea of code reusage was reported by S. Fimreite. WebEx also reuses components of the code

basis of EchoPAC, although this code is most probably not connected to UI definition, but fulfils the underlying functions.

The screenshots of the analysed products showed again that every company has another opinion on UI design. Thus, it is vital to focus on keeping the interface within the VP family similar as was specified in RS 7.2. Furthermore, Trium CTG Online already made some approaches to adjust their programme design to the one of VP, still RS 7.3 must also hold for future versions of both products.

RS 7.4 to support automatic testing of the Web client was not considered in the product comparison. Still, it is indisputable that this is a reasonable requirement. The proper operation of WebEx is also audited with the help of testing tools included in Microsoft Visual Studio .NET.

Handling image data carefully so that its medical significance and integrity is assured like specified by RS 7.5 has also proven to be a crucial issue. WebEx addressed this issue by offering two ways of distributing image data by means of the Web client. On one hand, image data is converted to Internet media formats for simple reviewing purposes. On the other hand, the original image data as acquired by the imaging modalities is provided for post-processing purposes.

RS 8.1 to keep the hardware requirements as simple as possible was not considered in the product comparison. Still, it is clear that this is a significant requirement to fulfil in order to have a real advantage over the Native client.

The analysed products did not connect directly to any hardware device. So it seems adequate to exclude any direct interfacing with hardware as proposed by RS 8.2.

6.3. Terminal Client as Alternative to a Web client

6.3.1. Terminal Server and VP

In section 2.7 the Terminal client-server architecture has been generally introduced as an alternative to a Native client or a Web client.

At first glance, one might think that a Terminal server could also be an equal solution to the issues addressed by the Web client. The motivation for considering a Terminal server in this thesis is based on the following idea. A Web client for VP cannot be offered ad hoc. A considerable amount of design and implementation work has to be done to create it. In contrast, by utilising the Terminal server as an alternative to the Web-based UI, no additional efforts are required. The Native client can be installed as it is on the Terminal server. Access to it is provided by the connected Terminal clients.

Following, it will be briefly described which experiences have been made in running the Native client of VP on Microsoft Server 2003 Terminal Services (MsTS2003). For details on VP provided by a Terminal server, please consult the related manuals or test protocols.

The Native client of VP has been tested extensively on the Terminal server MsTS2003 by VP-GE. There have not been any serious drawbacks, although certain limitations emerged.

As Terminal client only the latest Microsoft Remote Desktop Connection Client (RDC) running on Microsoft Windows has been tested. There is also an official version for Apple

Macintosh issued by Microsoft available. Furthermore, there are various implementations of Terminal clients on Linux available which can connect to MsTS2003 (corresponds to RS 2.1).

In general it can be stated that the Native client supports MsTS2003. All of the Native client's functions are provided to the connected Terminal clients (corresponds to RS 1.2). The following sites, for example, successfully access VP provided by MsTS2003 in everyday life:

- Maine Medical Center; Portland Maine; USA
- St. Vincent Hospital; Indianapolis Indiana; USA
- Frauenklinik, Kreiskrankenhaus Trostberg; Trostberg; Germany
- Krankenhäuser Siegmaringen, Pfullendorf und Saulgau; Germany
- Queensland Ultrasound for Womens; Brisbane; Australia

MsTS2003 can even connect to local hardware interfaces of the workstation running the RDC. Thereby, the measurement data transfer from US devices is supported. Moreover, the utilisation of locally connected Chip card readers, scanners and printers is supported by the Native client of VP running in the RDC. (corresponds to RS 8.2)

The only critical part when providing access to VP by the examined Terminal server is the viewing of image data, especially of video sequences. In general, the displayed colour depth of the examined Terminal client can be reduced to accelerate the performance via a low bandwidth connection. If this is done, image data reviewed at the corresponding Terminal client is displayed with a lower colour depth and, thus, looses its medical significance. This may lead to wrong decisions. Video sequences can only be viewed adequately by means of a high bandwidth connection. Otherwise the playback of sequences will no longer be smooth (corresponds to RS 7.5).

In a network performance test [Sch06] it appeared that working with VP is possible if a bandwidth of at least 64 kbit/s is offered. Data entry is possible although the operation of VP is not smooth. The reviewing of image data is impossible, though, as it is too slow. A broadband connection of 1 Mbit/s and above is sufficient to review still pictures. At least 2 Mbit/s are needed for video sequence replay (corresponds to RS 3.1 and RS 3.3).

Sudden connection loss, as it occurs in unstable networks, is not a problem for VP in case it is provided by the examined Terminal server. The started Terminal session and, therefore, the data entered in VP will be kept on the server until the user reconnects or the session expires. Also, an IP address change is no problem, as only the Terminal server has to cope with it and it does not affect VP in any way (corresponds to RS 3.2 and RS 4.6).

The user has to login via the examined Terminal client on the Terminal server, like a normal user sitting in front of the computer. Thereby, the means for authorisation and authentication of MsTS2003 are used (corresponds to RS 4.1 and RS 4.2). Moreover, as the Native client of VP is used, its means for authorisation, authentication and accountability are applied (corresponds to RS 4.3). In contrast, this is a fairly complicated procedure in which the user has to perform multiple logins until she/he is able to use VP (corresponds to RS 6.1 and RS 6.2). Additionally, MsTS2003, RDC and the Native client of VP do

not offer any means to synchronise the patient context apart from supporting CCOW [3] (corresponds to RS 6.3 and RS 6.4).

Since MsTS2003 SP1, the RDC can connect to the server over an SSL-encrypted connection. So means for Integrity control are also offered (corresponds to RS 4.4).

The protocol which is used for communication between MsTS2003 and RDC is called Remote Desktop Protocol (RDP). It utilises port 3389 by default, which is not a standard port (corresponds to RS 3.4).

The Terminal client RDC is by default included in Microsoft Windows XP. On other Microsoft Windows versions it can be added. It is a single executable file which does not have to be installed. In case a new version of RDC is issued, it can be exchanged. So the Terminal client demands a minimum of installation and maintenance efforts, apart from the setting up and administrating VP on the MsTS2003 (corresponds to RS 5.1 and RS 5.2).

As the Native client is provided by the Terminal server, no new UI will be created. Thus RS 7.1, RS 7.2, RS 7.3 and RS 7.4 are irrelevant.

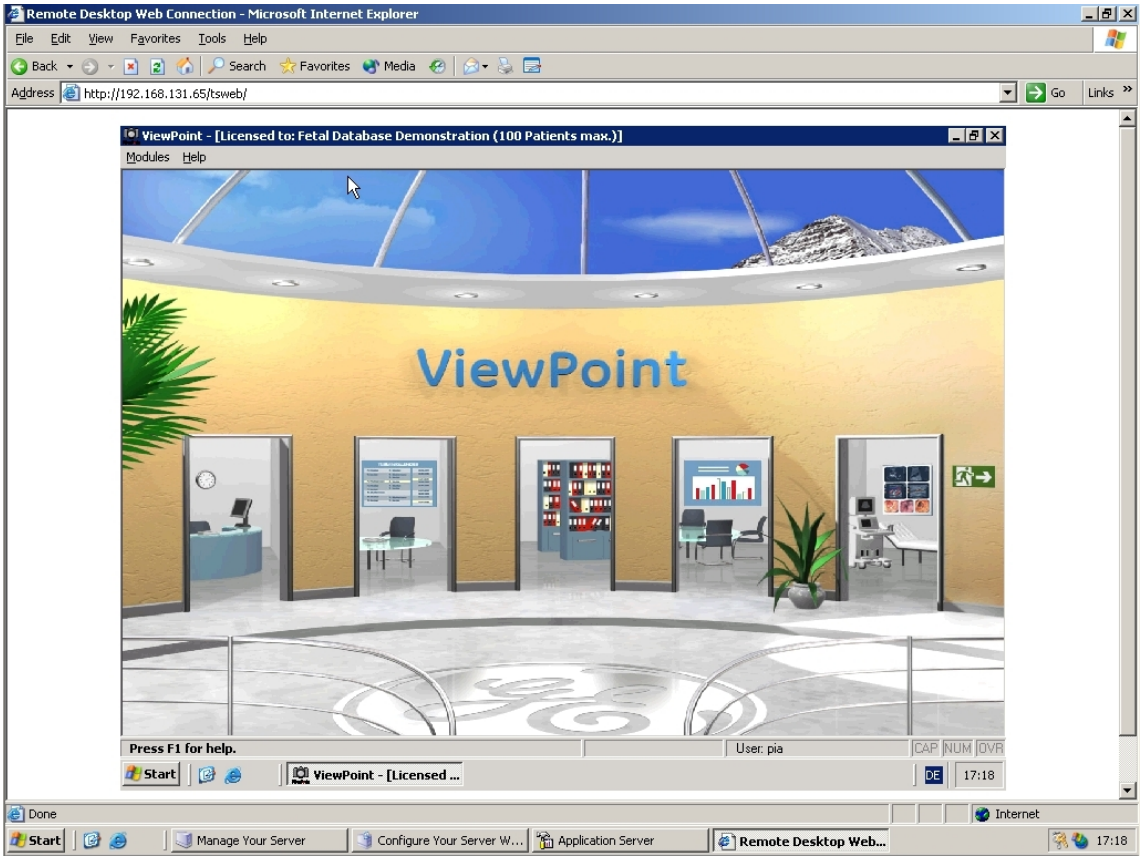


Figure 6.7.: Screenshot of ViewPoint's Native client running in the Microsoft Remote Desktop Web Connection.

[3] A Terminal server implementation of Citrix, which supports CCOW, also exists. It was not considered, though, as it would go beyond the scope of this thesis. For more information please refer to `http://www.citrix.com`, for example.

With the Microsoft Remote Desktop Web Connection (RDWC) it is even possible to offer a Web-based access to VP. This means that a Web browser can be started and a Web page which contains an embedded Terminal client similar to RDC can be opened. Until now this is restricted to Microsoft Internet Explorer only. With this client a connection to the Terminal server on which VP runs can be established. A screenshot of the RDWC with VP running in it is shown in figure 6.7. The RDWC is an ActiveX control and can be hosted by any Web server.

Limitations of the RDWC are that it does not support full colour depth and, as it is an ActiveX control, can only be used by workstations running Microsoft Windows and Microsoft Internet Explorer [4].

The description given here is only meant to give a brief idea of VP running on the Terminal server MsTS2003. Additionally, it will be presented to which extend this solution fulfils the requirements specification set up in section 5.3. A detailed analysis and evaluation would go beyond the scope of this thesis.

As mentioned above, the company Citrix offers also Terminal server solutions. They provide more sophisticated functionality like the support of CCOW or an optimised protocol for low bandwidth connections. Citrix's solutions should also be examined as alternative Terminal server. However, they were not considered in this thesis.

6.3.2. Web Client vs. Terminal Client

Following, the future Web client, implemented on the basis of the given requirements specification, will be compared to the Terminal client/server solution described in section 6.3.1. The considered Terminal server is MsTS2003. Both solutions will be compared on the basis of the specified requirements.

Table 6.2.: Comparison of Web Client vs. Terminal Client

Requirement	Web Client	Terminal Client
General requirements		
RS 1.1	fulfilled	fulfilled (via RDWC)
RS 1.2	fulfilled	fulfilled
RS 1.3	fulfilled	fulfilled
Requirements concerning supported Platforms, Web browsers and required Web browser extenions		
RS 2.1	fulfilled	fulfilled
RS 2.2	fulfilled	naturally only Microsoft Internet Explorer on Microsoft Windows is supported
RS 2.3	fulfilled	fulfilled (no extensions required)
Requirements concerning quality of supported networks		
RS 3.1	fulfilled	fulfilled

continued on next page

[4] There are also Plug-ins available which enable the use of ActiveX controls in Firefox. They were not considered, though, as it would go beyond the scope of this thesis.

Requirement	Web Client	Terminal Client
RS 3.2	partly fulfilled	fulfilled
RS 3.3	fulfilled	fulfilled
RS 3.4	fulfilled	RDP utilises port 3389 by default
Requirements concerning data security		
RS 4.1	fulfilled	fulfilled
RS 4.2	fulfilled	fulfilled
RS 4.3	fulfilled	fulfilled
RS 4.4	fulfilled	fulfilled
RS 4.5	*not examined in detail*	*not examined in detail*
RS 4.6	partly fulfilled	fulfilled
Requirements concerning installation and maintenance		
RS 5.1	fulfilled	fulfilled
RS 5.2	fulfilled	fulfilled
Requirements concerning interaction with other application components		
RS 6.1	fulfilled	not fulfilled
RS 6.2	fulfilled	not fulfilled
RS 6.3	fulfilled	not fulfilled
RS 6.4	fulfilled	not fulfilled
Requirements concerning the UI		
RS 7.1	fulfilled	irrelevant
RS 7.2	fulfilled	irrelevant
RS 7.3	fulfilled	irrelevant
RS 7.4	fulfilled	irrelevant
RS 7.5	fulfilled	not fulfilled when using low bandwidth connection or RDWC/ fulfilled when using high bandwidth connection and RDC
Requirements concerning hardware		
RS 8.1	fulfilled	fulfilled
RS 8.2	fulfilled	hardware control possible

According to the preceding comparison of the Web client and the Terminal client, both solutions have their advantages and disadvantages. From the current point of view it cannot be clearly determined which of them should be preferred.

Terminal client By picking the Terminal client, it would be possible to guarantee access integration while offering all of the functions of the Native VP client. This would also comprise the access and control of hardware which is, for example, required for the measurement data transfer. Unstable networks would also be supported by this solution, as a sudden connection loss would have no influence on the data already entered. Furthermore, by utilising the RDWC it would be possible to provide a Web-based UI without any remarkable design and implementation efforts like they are required for the Web client. Moreover the RDWC needs no additional Plug-ins.

Disadvantages of this solution are that the supported Web browsers are restricted to Microsoft Internet Explorer and Microsoft Windows. Moreover, the Terminal server utilises

a non-standard network port which might be blocked by the hospital's firewall. In respect to the Web client a comparatively high bandwidth connection is required to allow an adequate viewing of image data. Furthermore, the Terminal client does not offer any means to enable contextual integration.

Web client By choosing the Web client, naturally all of the requirements set up in section 5.3 would be fulfilled as it would be created on its basis. For example, contextual integration would be fully supported. Additionally, the Web client could also be used over connections with a low bandwidth. Furthermore, the Web client would also utilise standard network ports only.

The main disadvantage of the Web client is that it does not yet exist and a great amount of design and implementation work would be required to put the requirement specification into reality. Moreover, the functions it offers would be limited compared to those of the Terminal client, as it is not possible to control hardware via a Web-based UI. Furthermore, the support of unstable networks is a major issue and cannot be fully resolved by the Web client. Farther, it is inevitable that the Web client will utilise certain Web browser extensions.

7. Conclusion

7.1. Outline

In the preceding chapter several products offering Web-based UIs have been analysed and compared. Furthermore, the Terminal client-server architecture has been examined as an alternative to the Web client.

The aim of this chapter is to draw the conclusion of this thesis. In section 7.3 the thesis will be discussed in general. Section 7.3 will focus on examining to which extend the defined objectives have been achieved. Section 7.4 will view a short prospect on the subject of this thesis.

7.2. General Discussion

When the task of "Creating a Web client for VP" came up at VP-GE as potential topic for a diploma thesis, it was clear that it would offer work for several diploma theses. So the topic of this thesis was primarily limited to the development of a design concept for a Web client and the implementation of a prototype to demonstrate its viability. These two objectives had originally been accepted by VP-GE and IMISE as subject for this diploma thesis.

After creating the introduction and an initial literature research, it turned out that these objectives have still been too wide to be adequately covered in the scope of a diploma thesis. Furthermore, it was later demanded by VP-GE and IMISE that the Terminal client-server architecture should also be partly covered by this diploma thesis as alternative technology to the Web client.

Finally, VP-GE and IMISE agreed to alter the original objectives in the way they are defined now (see section 1.4).

In order to achieve these objectives, the author originally intended to answer the questions and to follow the related task definitions subsequently as specified in section 1.5. As the question and task definition has, however, been created at the beginning, when the author had not yet acquired a broad understanding of the problem domain and the connected efforts, it soon turned out that it was obsolete to subsequently perform the tasks. For example, the first part of Question 3 ("How is VP integrated within the HIS at the moment?") had to be answered before being able to answer Question 1 ("Which requirements should be fulfilled by the Web-based UI?"). Furthermore, question 5 and 6 had to be shortened, as it was impossible to set up different concepts for a Web client in the given period of time. It was agreed with VP-GE and IMISE to only compare the features of a Web client to those of a Terminal client and to determine to which extend they fulfil the requirement specification set up in this thesis.

Still, during creation of this thesis the question and task definition has been used as an orientation. As the thesis' subject and scope are officially defined by the objectives, the conclusion will be drawn based on these objectives. In the succeeding section it will be reasoned why these objectives have been successfully achieved.

In general, it emerged in the course of this thesis, that it would have been more appropriate to start from another point of view by defining the objectives and the general subject of this thesis in a more abstract way. One of the main reasons for hospitals to demand for a Web client is to have an application component which enables access integration and generally offers adequate integration capabilities. Thus, it would have been more appropriate to name the topic of the corresponding thesis something like "Improve the degree of integration of a medical documentation and image archiving system within hospital information systems". Furthermore, a potential objective would be something like "Establish a recommendation for what technology should be used to enable access integration". So, the Terminal client-server technology could have been equally evaluated. Also, further technologies or methods as alternatives to the Web-based access could have been considered.

However, this is not the case. Many hospitals expressly demand for a Web client in their tenderings. VP-GE wants to serve them and thereby needs adequate means to provide Web-based access. VP-GE offered this thesis with the intention to gain ideas and concepts about how to implement a Web client. So right from the start, it was clear that a potential Web client is the focus of interest in this thesis, no matter if it is the best technique to offer adequate integration capabilities or not.

7.3. Achievement of Objectives

In section 1.4 the objectives which should be covered by this diploma thesis have been defined. Following, it will be verified if these objectives have been achieved in the course of the thesis.

7.3.1. Objective 1

> *Determine which requirements are not met by the Native client of VP and the VP system in general for an adequate integration within the HIS.*

This objective has been fully achieved.

In section 3.4.2 it was examined to what degree VP and its Native client support the different types of integration within the HIS.

- Data integration is supported to a high degree by VP. The only weak point is that a linkage between VP-Reports and the corresponding image data is not supported once they have been exported to other application components (WP 10). See section 3.4.2.2 for details.
- Access integration is not guaranteed by VP. This is due to the special characteristics of the Native client which cannot be installed wherever access to VP is needed (WP

1 to WP 6). Furthermore, VP does not offer any other means to provide location-independent access, like a Web-based UI (WP 9). See section 3.4.2.2 for details.

- Presentation integration between VP and other application components is not possible. This is because no common guidelines or standards which define the layout and handling of the UI in a unique way exist.

  Presentation integration within the different reporting modules of VP is guaranteed (WP 12).

  Presentation integration is not guaranteed between VP and Trium CTG Online, which are sold together to obstetrical departments as an all-in-on solution (WP 13).

  See section 3.4.2.3 for details.

- Contextual integration is not supported by VP as it offers no means to synchronise patient or user context with other application components (WP 14). See section 3.4.2.4 for details.

7.3.2. Objective 2

Establish a recommendation for what technology should be used to implement a Web client.

With the Web client access to VP should be possible from every workstation which is networked with the VP server, without any additional application components, and by using a standard Web browser.

Limitations: A selection of software products with a Web-based UI should be examined and described to also consider existing solutions during the creation of the concept. Selection is made in accordance with VP-GE and IMISE.

Terminal server technology should also be examined as an alternative technique to the Web-based access to VP.

This objective has been achieved, with the limitation that the recommendation is left on a quite abstract level.

Requirement specification In order to be able to give a recommendation for technologies which should be used to implement a Web client, first of all, the Web client's characteristics and desired features had to be determined. This was done by setting up a requirements specification. Details of the approach used for requirements engineering can be found in section 2.3.

First, a requirements analysis was performed in chapter 3. The resulting requirements were generally defined in chapter 4. The specification of requirements was realised in chapter 5. Following, the created requirements specification will be shown. Certain requirements have been shortened, indicated by "[...]", for the purpose of enabling a quick overview. The complete requirements and details on them can be found in section 5.3.

Table 7.1.: Short Overview of the requirements for a Web Client

	Requirement
General requirements	
RS 1.1	VP shall provide a Web client.
RS 1.2	The Web client shall provide all functions of a Native client [...], apart from the control and access of hardware[...].
RS 1.3	The Web client shall provide access to VP from local and remote locations.
Requirements concerning supported Platforms, Web browsers and required Web browser extenions	
RS 2.1	The Web client shall support Microsoft Windows XP, Microsoft Windows 2000 and Linux on the Intel *x86* hardware architecture.
RS 2.2	The Web client shall support Firefox [...] and Microsoft Internet Explorer [...].
RS 2.3	The Web client shall utilise Web browser extensions to support the display of image data and to fulfil RS 3.1 and RS 3.3. [...]
Requirements concerning quality of supported networks	
RS 3.1	The Web client shall keep the bandwidth utilisation as low as possible.
RS 3.2	The Web client shall also support unstable networks.
RS 3.3	The Web client shall perform and respond in such a manner that the user is not hindered in doing her/his actual work.
RS 3.4	The Web client shall utilise only standard ports like `HTTP` port 80 or `HTTPS` port 443.
Requirements concerning data security	
RS 4.1	The Web client shall assure secrecy and confidentiality by offering sufficient means for authorisation.
RS 4.2	The Web client shall offer sufficient means for authentication.
RS 4.3	The Web client shall offer sufficient means for non-repudiation and accountability.
RS 4.4	The Web client shall offer sufficient means for integrity control.
RS 4.5	The technologies chosen to address RS 4.1, RS 4.2, RS 4.3 and RS 4.4 shall fulfil all legal regulations [...].
RS 4.6	The Web client shall offer means to protect against data loss.
Requirements concerning installation and maintenance	
RS 5.1	The Web client shall require no or at least a minimum of installation and configuration efforts.
RS 5.2	The Web client shall require no or at least a minimum of maintenance efforts.
Requirements concerning interaction with other application components	
RS 6.1	The Web client shall offer means to switch from itself to another application component [...] without having to reenter login information.
RS 6.2	The Web client shall offer means to switch from another application component [...] to itself without having to reenter login information.
RS 6.3	The Web client shall offer means to switch from itself to another application component [...] without having to reenter patient identification information to preserve the patient context.
RS 6.4	The Web client shall offer means to switch from another application component [...] to itself without having to reenter patient identification information to preserve the patient context.
Requirements concerning the UI	

continued on next page

	Requirement
RS 7.1	The UI of the Web client shall be based on the existing masks definition if possible and feasible [...]. If not possible and feasible, the Web client shall utilise a standardised UI definition language [...].
RS 7.2	The Web client shall offer a UI which resembles the Native client in its appearance and is controlled similarly.
RS 7.3	The UI of the Web client shall offer [...] similarity to the UI of Trium CTG Online [...].
RS 7.4	The Web client shall offer a UI which supports automatic testing.
RS 7.5	The Web client shall preserve the quality and integrity of all image data provided to the user via the Web-based UI.
Requirements concerning hardware	
RS 8.1	The Web client shall keep its hardware requirements to a minimum.
RS 8.2	The Web client shall offer no means to control hardware devices connectable to VP.

Technology recommendation During requirement specification it already turned out that it would be impossible to give a detailed technology recommendation for the implementation of the Web client. For a serious recommendation, the available technologies for the realisation of each requirement would have to be carefully analysed and evaluated in detail. If this had been done for all requirements, the connected efforts would have gone far beyond the scope of this thesis.

Still, during requirement specifications wherever possible it was tried to give recommendations which technologies could be used or at least which procedure should be preferred to fulfil the related requirement. Outstanding recommendations which are not directly defined by the requirement specification, for example that the Web client should support certain Platforms or Web browsers versions, are:

- The most adequate Plug-ins to enable viewing of image data through the Web client are Java Applets like introduced in 5.2.4. See section 5.2.4 for details.
- In order to reduce the network traffic and to enable immediate response to certain user activities, a Plug-in, for example Java Script, should be used to be able to edit the related Web page directly without requiring communication with the server. See section 5.2.4 for details.
- For authorisation and authentication purposes the potentials of SSO solutions should be considered.
- For authentication purposes the potentials of the HPC should be considered.
- In order to prevent data loss during data entry, data entered already should be automatically saved on the server at regular intervals.
- In order to enable contextual integration CCOW should be the preferred standard. It also offers means for authorisation and authentication. See section 5.2.5 for details.
- In order to enable the creation of different UIs, from one definition, like required for the Native and Web client, `XML`-based Markup languages for UI definition can be potential candidates. See RS 7.1 for details.

Product analysis In order to determine how other manufacturers have realised a Web-based UI and if, compared to their solutions, the requirement specification set up in this thesis is adequate, an analysis and comparison of several products was performed. The product comparison is intended to be used as decision support for the later implementation of the requirement specification. It covers nearly every requirement and is structured in a similar way (see 6.2.4 for details on the comparison criteria).

The following products have been examined:

- E3 of Euroking Miracle Ltd.. See section 6.2.1 for details.
- WebEx of GE Vingmed Ultrasound. See section 6.2.2 for details.
- Trium CTG Online of Trium Analysis Online GmbH. See section 6.2.3 for details.

The comparison of these products can be found in table 6.1 on page 95.

It turned out that the requirement specification set up is adequate and should be used to implement the Web client. Details on the justification of this statement on the basis of the product comparison can be found in section 6.2.6.

Terminal server As VP is already provided through a Terminal server at some customer sites, the Terminal client-server architecture was considered as possible alternative to the Web client. This was further encouraged by the availability of a Web interface for the Terminal client (RDWC).

An evaluation of the Terminal server/client technology and its applicability to realise the requirements specification was performed in section 6.3.1. In table 6.2 on page 101 it was analysed to which extend it fulfils the requirements in comparison to the Web client .

It turned out that the Terminal client could be a serious alternative to the Web client, especially when using the RDWC.

Outstanding advantages are that it could be used ad hoc without requiring any additional design or implementation efforts. Moreover the access and control of hardware are supported by the examined Terminal client. Still, certain drawbacks are linked to its usage. For example, does RDWC only supports Microsoft Windows and Microsoft Internet Explorer. Also, the examined Terminal client does not offer any means to enable contextual integration. See section 6.3.2 for details.

7.3.3. Objective 3

> *The degree of integration of VP in the HIS should be analysed and modelled by using the $3LGM^2$ meta model. The current integration degree with the Native client and the future integration degree with the additional Web client should be examined.*

This objective was fully achieved.

In section 3.4 the current degree of integration of VP within the HIS was modelled by the utilisation of the $3LGM^2$ meta model. Figure 3.12 on page 57 shows the related logical tool

layer, while figure 3.13 on page 58 shows the physical tool layer. During the identification of the weak points of VP in respect to integration, the model has been used.

In section 4.3 the future degree of integration of VP was modelled. Figure 4.1 shows the related logical tool layer, figure 4.2 shows the physical tool layer. The aim was to illustrate to what extend the future Web client would increase the degree of integration of VP.

7.4. Prospect

In summary the Web client turned out to be an adequate approach for integrating the medical documentation and image archiving system VP within the HIS. Apart from, the interfaces for data integration already offered by VP, the Web client is the preferable solution to provide the missing means for access integration and contextual integration.

If the Terminal client could be an alternative to this approach has to be researched in more detail.

As far as it can be judged from the point of view of this thesis, the Terminal client solution and its Web-based UI could be utilised to bridge the time gap until the Web client is implemented and officially released.

Bibliography

[Alb03] Albrecht, C. C.: A Comparison of Distributed Groupware Implementation Environments. In: *36th Annual Hawaii International Conference on System Sciences (HICSS'03)* (2003)

[AMCBS04] Azevedo-Marques, P.M. de ; Carita, E.C. ; Benedicto, A.A. ; Sanches, P.R.: Integrating RIS/PACS: The Web-based Solution at University Hospital of Ribeirao Preto, Brazil. In: *Journal of Digital Imaging* 17 (2004), Nr. 3, S. 226–233

[Bal98] Balzert, H.: *Lehrbuch der Software-Technik: Software-Management, Software-Qualitaetssicherung, Unternehmensmodellierung.* Spektrum Akademischer Verlag, 1998. – ISBN 3827400651

[BK02] Bunse, C. ; Knethen, A. von: *Vorgehensmodelle kompakt.* Spektrum Akademischer Verlag, 2002. – ISBN 382741203X

[DH94] Davis, A. M. ; Hsia, P.: Giving Voice to Requirements Specification. In: *IEEE Software* (1994), S. 12–16

[EMSM04] Engelmann, U. ; Münch, H. ; Schroeter, A. ; Meinzer, H.P: CHILI/Web: Klinikweite Bildverarbeitung aus der elektronischen Patientenakte heraus. In: *Telemedizinführer Deutschland* 2004 (2004)

[ESM+05] Engelmann, U. ; Schroeter, A. ; Münch, H. ; Schweitzer, T. ; Braun, V. ; Meinzer, H.P: Von der bilateralen Teleradiologie zur Vernetzung von Regionen - der CHILI-Ansatz. In: *Telemedizinführer Deutschland* 2005 (2005), S. 261–265

[Gie06] Giesecke & Devrient GmbH: *The Health Professional Card (HPC).* Version: 2006. `http://www.gi-de.com/portal/page?_pageid=42,55030&_dad=portal&_schema=PORTAL`, Last checked: 2 May 2006

[HA05] Haux, R. ; Ammenwerth, A.: *IT-Projektmanagement in Krankenhaus und Gesundheitswesen.* 3794524160, 2005. – ISBN 3794524160

[HAWB04] Haux, R. ; Ammenwerth, E. ; Winter, A. ; Brigl, B.: *Strategic Information Management in Hospitals.* Springer, 2004 (Health Informatics). – ISBN 0387403566

[HBABW05] Hübner-Bloder, G. ; Ammenwerth, E. ; Brigl, B. ; Winter, A.: Specification of a Reference Model for the Domain Layer of a Hospital Information System. In: *Studies in Health Technology and Informatics* 116 (2005), S. 497–502

[Hol91] Holzmann, G. J.: *Design and Validation of Computer Protocols*. Prentice-Hall, 1991. – ISBN 0135399254

[KK03] Kroll, P. ; Kruchten, P.: *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. Addison Wesley, 2003. – ISBN 0321166094

[LGH+03] Leiner, F. ; Gaus, W. ; Haux, R. ; Knaup-Gregori, P. ; Pfeiffer, K.-P.: *Medizinische Dokumentation*. Schattauer, 2003. – ISBN 3794522656

[Mat04] Mathers, T. W.: *Windows Server 2003/2000 Terminal Server Solutions*. 4. Addison Wesley Professional, 2004. – ISBN 1578702763

[MEAM03] Münch, H. ; Engelmann, U. ; A.Schroeter ; Meinzer, H.P.: Web-based distribution of radiological images from PACS to EPR. In: *International Congress Series* 1256 (2003), S. 873–879

[Neo06] NeoLogica s.r.l.: *RemotEye_big_1.jpg*. Version: 2006. `http://eng.neologica.it/prodotti/remoteye/images/RemotEye_big_1.jpg`, Last checked: 7 April 2006

[Psc02] Pscheidl, K. ; ViewPoint Bildverarbeitung GmbH – GE Healthcare (Hrsg.): *PIA 3.31 Specification*. 0.3. Wessling: ViewPoint Bildverarbeitung GmbH – GE Healthcare, 02 2002

[Rie05a] Rieger, G. ; ViewPoint Bildverarbeitung GmbH – GE Healthcare (Hrsg.): *System Requirements for ViewPoint 5.0*. 3.0. Wessling: ViewPoint Bildverarbeitung GmbH – GE Healthcare, 09 2005

[Rie05b] Rieger, G. ; ViewPoint Bildverarbeitung GmbH – GE Healthcare (Hrsg.): *ViewPoint Basic User Manual*. 1.0. Wessling: ViewPoint Bildverarbeitung GmbH – GE Healthcare, 09 2005

[RJB99] Rumbaugh, J. ; Jacobson, I. ; Booch, G.: *The Unified Modeling Language – Reference Manual*. Addison Wesley, 1999. – ISBN 020130998X

[Rus06] Rusak, A.: *ViewPoint 5.01 DICOM Conformance Statement*. 0.1. Wessling, 01 2006

[Sch06] Schmuhl, H.-H. ; ViewPoint Bildverarbeitung GmbH – GE Healthcare (Hrsg.): *Terminal Server Performance Test*. 1.0. Wessling: ViewPoint Bildverarbeitung GmbH – GE Healthcare, 03 2006

[Seh05] Seharsch, A. ; ViewPoint Bildverarbeitung GmbH – GE Healthcare (Hrsg.): *SRS ViewPoint 5.0 – Core and Application Requirements*. 1.0. Wessling: ViewPoint Bildverarbeitung GmbH – GE Healthcare, 09 2005

[Sel01] Seliger, R.: *Overview of HL7's CCOW Standard*. Version: 2001. `http://www.hl7.org/library/committees/sigvi/ccow_overview_2001.doc`, Last checked: 17 May 2006

[Sen00] Sentillion: *CCOW Tutorial*. Version: 2000. `http://www.icis.ca/cihiweb/en/downloads/infostand_hl7ehealth_may01_e_Sentillion_ccow.pdf`, Last checked: 17 May 2006

[Som95] SOMMERVILLE, I.: *Software Engineering*. 5. Addison Wesley, 1995. – ISBN 0201427656

[Tan03] TANENBAUM, Andrew S.: *Computer Networks*. 4. Pearson Education, Inc., 2003. – ISBN 0130384887

[The06] THECOUNTER.COM: *Main Stats*. Version: 2006. `http://www.thecounter.com/stats/2006/March`, Last checked: 4 April 2006

[Ups06] UPSDELL, C. A.: *Browser News: Statistics*. Version: 2006. `http://www.upsdell.com/BrowserNews/stat.htm`, Last checked: 4 April 2006

[W3S06] W3SCHOOLS.COM: *Browser statistics*. Version: 2006. `http://www.w3schools.com/browsers/browsers_stats.asp`, Last checked: 4 April 2006

[WBW03] WINTER, A. ; BRIGL, B. ; WENDT, T.: Modeling Hospital Information Systems (Part 1): The Revised Three-layer Graph-based Meta Model $3LGM^2$. In: *Methods Inf Med* 42 (2003), S. 544–51

[WHBW04] WENDT, T. ; HÄBER, A. ; BRIGL, B. ; WINTER, A.: Modeling Hospital Information Systems (Part 2): Using the $3LGM^2$ Tool for Modeling Patient Record Management. In: *Methods Inf Med* 43 (2004), S. 256–67

[Wik06] WIKIPEDIA, THE FREE ENCYCLOPEDIA: *Interview - Wikipedia, the free encyclopedia*. Version: 2006. `http://en.wikipedia.org/wiki/Interview`, Last checked: 1 April 2006

[Wil03] WILLIAMS, T. N.: *Proposal for CCOW Specification - Version 0.1*. 2003

Index

3LGM^2, 5, 6, 9, 14, 16–19, 54, 66, 108
.NET, 86

ActiveX, 98, 99
Adobe Acrobat Reader, 25, 84, 91, 93
Apache, 91, 93
Apple Macintosh, 55, 73, 74, 88, 93, 96
application component, 1–6, 10–12, 16–18, 22, 27–30, 32, 33, 35, 37–39, 49, 52, 53, 55, 56, 59, 63, 65–67, 71, 72, 75, 77, 78, 81, 84–87, 90, 95, 104–106
Application Component Configuration, 18
Application programme, 17
ASCII, 37
ASP.NET, 93
AVI, 44

BDT, 3, 10, 49, 50
BMP, 44

Cardiotocography, 90, 91
CCOW, 3, 11
CDMS, 9
Central Processing Unit, 27, 32
CHILI/Web-Viewer, 75
Citrix, 97, 99
Clinical Context Object Workgroup, 3, 4, 9, 11, 28–30, 59, 66, 67, 77, 79, 81, 86, 89, 91, 93–95, 97, 99, 107
Clinical Documentation and Management System, 9, 39, 41, 49, 56, 72, 75, 87, 90
Clinicom, 11
clinicom, 49, 50
Cloverleaf, 49
Common Object Request Broker Architecture, 52, 55
Cookies, 60
CORBA, 52
CPU, 27
CTG, 90

Data Processing Component Configuration, 18
Database Management System, 31, 86, 91, 92
dBASE, 37
DBMS, 31
DICOM, 2, 3, 11, 31, 35, 38, 40, 44, 50–53, 70, 75, 76, 84, 87
DICOM Viewer, 75
Digital Imaging and Communications in Medicine, 11

E3, 21, 22, 83–86, 93–95, 108
EchoPAC, 21, 87, 88, 95
Enterprise Architect, 16
Enterprise function, 16, 17, 54, 67
Entity types, 17
Euroking Miracle Ltd., 21, 22, 83, 86, 93, 108
EViewBox, 75
Extensible Application Markup Language, 81
Extensible HyperText Markup Language, 88, 93
Extensible Markup Language, 30, 81, 84, 107
Extensible User Interface Language, 81

Fetal Medicine Foundation London, 37, 54
Firefox, 4, 24, 59, 73–75, 78, 91, 93, 99, 106
Firewall, 60, 61, 64
FMF, 37
Framegrabber, 43, 44, 52, 61, 70

G Web server, 93
GE Healthcare, 83, 87, 90
GE Vingmed Ultrasound, 21, 22, 86–89, 93, 108
GIF, 93

Hard Disk Drive, 32
HCM, 11, 49, 50

HDD, 32
Health Insurance Portability and Accountability Act, 10, 28, 80
Health Level 7, 3, 5, 28, 49, 50, 77, 84, 87, 90
Health Level Seven, 11
Health Professional Card, 79, 80, 86, 88, 91, 95, 107
Helper Application, 25, 60, 64, 74, 75
HIPAA, 10, 28
HIS, 2
HL7, 3, 11
Hospital Information System, 2–6, 9, 10, 16, 18, 31, 54, 66, 104, 108, 109
HPC, 79
HTML, 24
HTTP, 30
HTTPS, 64
Hypertext Markup Language, 24, 25, 78, 91–93
HyperText Transfer Protocol, 30, 60, 64, 67, 79, 93, 106
Hypertext Transfer Protocol Secure, 64, 79, 88, 91, 93, 106

IDE, 88
IIS, 84
image data, 1–4, 9, 32, 33, 37, 38, 40, 43–45, 50–52, 55–57, 61, 63, 66, 70, 75, 78, 82, 87, 88, 94, 96, 97, 100, 104, 106, 107
IMISE, 5
Institute for Medical Informatics, Statistics and Epidemiology, 5, 9, 16, 19, 103, 105
Integrated Development Environment, 88
Integrated Services Digital Network, 88
Intel, 9, 74, 78, 106
Inter-Layer-Relationships, 18
Internet, 4, 59, 60, 73, 75, 77, 88, 93, 95, 96
Intersystems Cache, 86, 93
Intranet, 4, 90, 93–95
ISDN, 88

Java, 25
Java Applet, 75, 76, 107
Java Script, 25, 60, 77, 84, 91, 93, 107
Java Virtual Machine, 84, 91, 93
JPEG, 44, 75, 88, 91, 93
JVM, 84

LAN, 23
Linux, 27, 55, 73, 74, 78, 88, 93, 96, 106
Local Area Network, 23, 32, 55, 59, 85, 95
Logical Tool Layer, 17
LOGIQworks FX, 45, 70, 76

Macromedia ColdFusion, 91, 93
Macromedia Flash, 25, 91, 93
MDS, 1
Medical Documentation System, 1, 32, 83, 93
Microsoft, 73, 78, 81, 84, 88, 94, 96
Microsoft Access, 93
Microsoft Active Server Pages .NET, 93
Microsoft DirectX, 31
Microsoft Excel, 37
Microsoft Internet Explorer, 24, 59, 73–75, 78, 84, 88, 91, 93, 94, 98–100, 106, 108
Microsoft Internet Information Service, 84, 85, 88, 93
Microsoft Remote Desktop Connection Client, 96–98
Microsoft Remote Desktop Web Connection, 98–100, 108
Microsoft Server 2003 Terminal Services, 96–99
Microsoft SQL Server, 93
Microsoft Visual Studio .NET, 88, 96
Microsoft Windows, 9, 27, 31, 44, 53, 55, 59, 73, 74, 78, 84–86, 88, 91, 93, 94, 96, 98–100, 106, 108
Microsoft Windows Fax Server, 45
Microsoft Word, 84, 93
Mozilla, 91, 93
Mozilla Foundation, 81
MPEG-1, 88, 93
MPEG-4, 88, 93
MsTS2003, 96

National Health Service, 84, 85, 94
Native Client, 23
Netscape, 88, 93
NHS, 84

ODBC, 49
Open DataBase Connectivity, 49, 50, 84, 87
Open Source, 75
OpenSA, 91, 93
Opera, 73, 88, 93

Operating System, 9, 15, 23, 24, 27, 64, 71–74, 78, 84, 94
OS, 9

PACS, 1
Patient Data Management System, 83, 90, 93
patient demographics, 2, 3, 29, 33, 35, 38, 49, 51–53, 56, 75, 90
Patient Management System, 2, 9, 35, 40, 49, 52, 56, 72, 75, 87, 90
PDF, 93
PDMS, 83
Physical Tool Layer:, 17
Picture Archiving and Communication System, 1, 32, 40, 50, 51, 56, 75, 87, 93
Platform, 9, 15, 23–25, 55, 60, 64, 71–74, 78, 88, 94, 99, 106, 107
Plug-in, 25, 45, 60, 64, 74, 77, 79, 99, 100, 107
PMS, 2
PNG, 44
Pop-Up, 60, 88
Portable Document Format, 93

Qt, 23

RAM, 27, 32
RDC, 96
RDP, 98
RDWC, 98
Remote Desktop Protocol, 98, 99
RemoteEye, 76

SAP IS-H*med, 28
SAP Remote Function Call, 11
SAP-RFC, 11, 49, 50
Secure Socket Layer, 30, 91, 93–95, 98
Service Pack, 31, 55, 64, 74
Simple Object Access Protocol, 30
Single Sign-On, 28, 77–79, 81, 85, 86, 107
SOAP, 30
Software product, 17
Sparx Systems Pty Ltd., 16
SPSS, 37
SSL, 30
SSO, 28
Sun Solaris, 74
Sybase SQL, 31

TCP/IP, 32
Terminal Client, 26
Terminal server, 5, 6, 10, 26, 27, 96–100, 105, 108
TIF, 44
Trium Analysis Online GmbH, 21, 22, 58, 82, 89–91, 93, 108
Trium CTG Online, 21, 22, 38, 58, 66, 82, 89–91, 93–96, 105, 107, 108
TWAIN, 44, 53, 70

UI, 3
Ultrasound, 43–45, 51–53, 87, 97
UML, 14
Uniform Resource Locator, 4, 24, 25, 30, 77, 81, 86, 92, 95
Universal Modelling Language, 14–16
URL, 4
US, 43
User Interface, 3–7, 9, 10, 15, 17, 22–24, 26–28, 43, 46, 49–51, 56, 58, 59, 61, 66, 67, 69, 71, 75, 77, 81–83, 88, 91, 92, 94–96, 98, 100, 101, 103, 105–109

ViewPoint, 1–7, 13, 16, 20, 22, 27, 28, 31–33, 35, 37–40, 43, 45, 46, 49–61, 63–67, 69, 71, 72, 75, 77, 78, 82, 83, 90, 94–100, 103–109
ViewPoint Bildverarbeitung GmbH - GE Healthcare Technologies, 1, 5, 6, 13, 16, 19, 20, 22, 49, 58, 61, 69, 82, 90, 96, 103–105
ViewPoint Report, 39, 41, 45, 53, 56, 57, 63, 70, 71, 104
Vivid, 87
Voluson 4D View, 45, 70, 76
VP, 1
VP-GE, 1
VP-Report, 39
VPAdmin, 46

WAN, 4
Web browser, 4, 5, 15, 24–26, 59–61, 64, 71–78, 80, 82, 84, 88, 91, 92, 94, 98–101, 105–107
Web Client, 24
Web server, 24
WebEx, 21, 22, 86–89, 93–96, 108
Wide Area Network, 4
Wireless Local Area Network, 55, 64

WLAN, 55

X Windows System, 27
XAML, 81
XHTML, 88
XML, 30
XUL, 81

ZIP, 88, 89, 93

A. Appendix

Table A.1.: Enterprise functions originating from the reference model of the domain layer proposed by Hübner-Bloder et al. [HBABW05] which are supported by the corresponding modules of ViewPoint.

Enterprise Function			VP Module
1. Patientenbehandlung	**1.1. Patientenaufnahme**	1.1.1 & 2.2.1 Vormerkung und Einbestellung von Patienten	-
		1.1.2 & 3.1.2 Identifikation und Prüfung auf Wiederkehrer	Patient Management Module
		1.1.3 & 3.1.1 Administrative Aufnahme	Patient Management Module
		1.1.4. Ärztliche Aufnahme	Patient Management Module, Case and Examination Management Submodule
		1.1.5. Pflegerische Aufnahme	-
		1.1.6 & 3.1.3 Patientenauskunft und Informationsdienste	-
	1.2. Entscheidungsfindung, Behandlungsplanung und -organisation		-
	1.3. Leistungsanforderung	**1.3.1 Erstellung eines Auftrages für eine Massnahme**	Examination Module
		1.3.2. Terminanforderung	-
	1.4. Diagnostische, therapeutische oder pflegerische Maßnahmendurchführung	**1.4.1 Diagnostische & therapeutische Massnahmendurchführung**	Examination Module
		1.4.2 Pflegerische Massnahmendurchführung	-
	1.5. Leistungsdokumentation	**1.5.1 Diagnosen-Klassierung**	Examination Module
		1.5.2 Massnahmen-Klassierung	Examination Module
	1.6. Entlassung und Weiterleitung an eine andere Einrichtung	1.6.1 & 3.1.4 Administrative Entlassung und Leistungsabrechnung	-
		1.6.2 Ärztliche Entlassung und Arzbriefschreibung	Examination Module
		1.6.3 Pflegerische Entlassung & Erstellung des Pflegeabschlussberichtes	

continued on next page

Enterprise Function			VP Module
2. Versorgungsmanagement, Ressourcenplanung und Arbeitsorganisation	2.1 Ver- & Entsorgungsmanagement		-
	2.2 Termin- und Ressourcenplanung	1.1.1 & 2.2.1 Vormerkung und Einbestellung von Patienten	-
		2.2.2 Termin- und Ressourcenplanung mit dem medizinischen Dienstleister	Patient Scheduling Module
		2.2.3 Termin- und Ressourcenplanung mit dem Patiententransportdienst	-
	2.3 Personalwesen		-
3. Krankenhausverwaltung	**3.1 Patientenverwaltung**	**3.1.1 & 1.1.3 Administrative Aufnahme**	Patient Management Module
		3.1.2 & 1.1.2 Identifikation und Prüfung auf Wiederkehrer	Patient Management Module
		3.1.3 & 1.1.6 Patientenauskunft und Informationsdienste	-
		3.1.4 & 1.6.1 Administrative Entlassung und Leistungsabrechnung	-
	3.2 Archivierung von Patienteninformationen	**3.2.1 Anlegen der Patientenakte**	Archive Module, Examination Module
		3.2.2 Verwaltung und Bereitstellung der Patientenakten	Archive Module, Examination Module
	3.3 Qualitätsmanagement	**3.3.1 Internes Qualitätsmanagement**	Statistics Module
		3.3.2 Erfüllung gesetzlicher Meldepflichten	Statistics Module, Rule-Checking Submodule
	3.4 Betriebssteuerung (Controlling)		-
	3.5 Kosten- und Leistungsrechnung		-
	3.6 Finanzbuchhaltung		-
	3.7 Gebäude- und Flächenmanagement		-
	3.8 Informationsmanagement		-
4. Krankenhausleitung			-
5. Forschung und Lehre			-
6. Sonstige Aufgaben			-

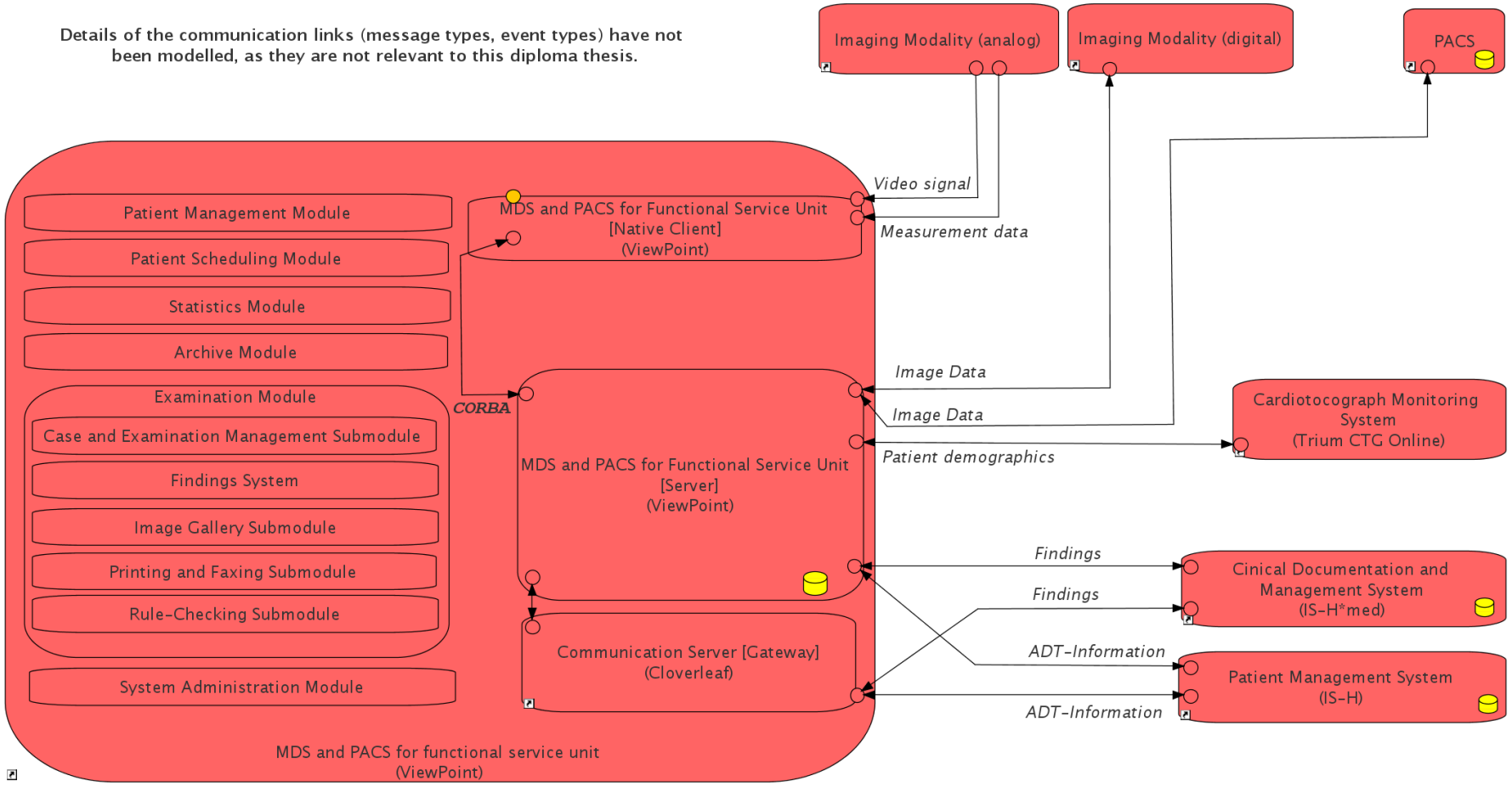


Figure A.1.: Logical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system.

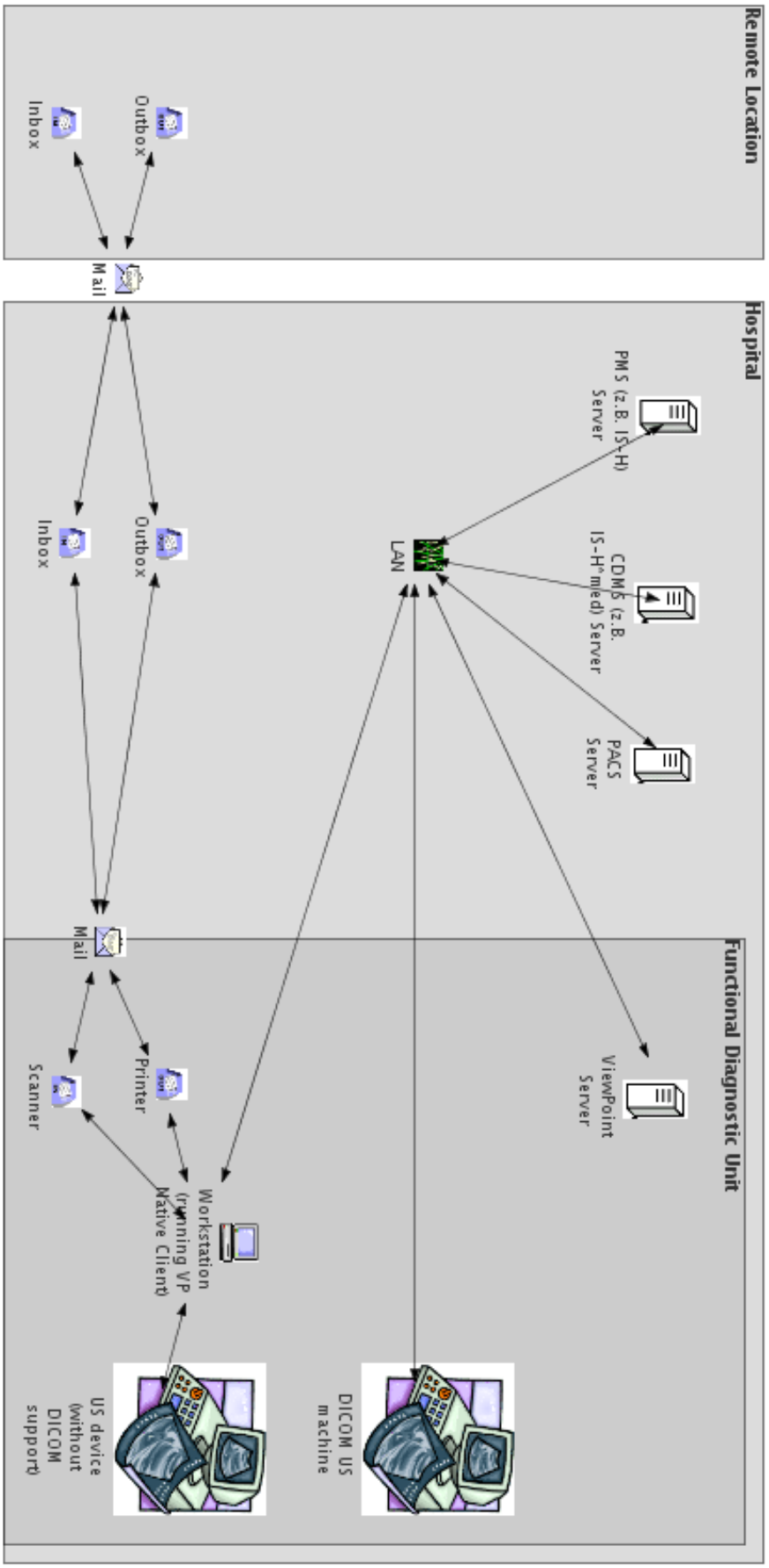


Figure A.2.: Physical tool layer of the 3LGM^2 model showing the current integration degree of ViewPoint within the hospital information system.

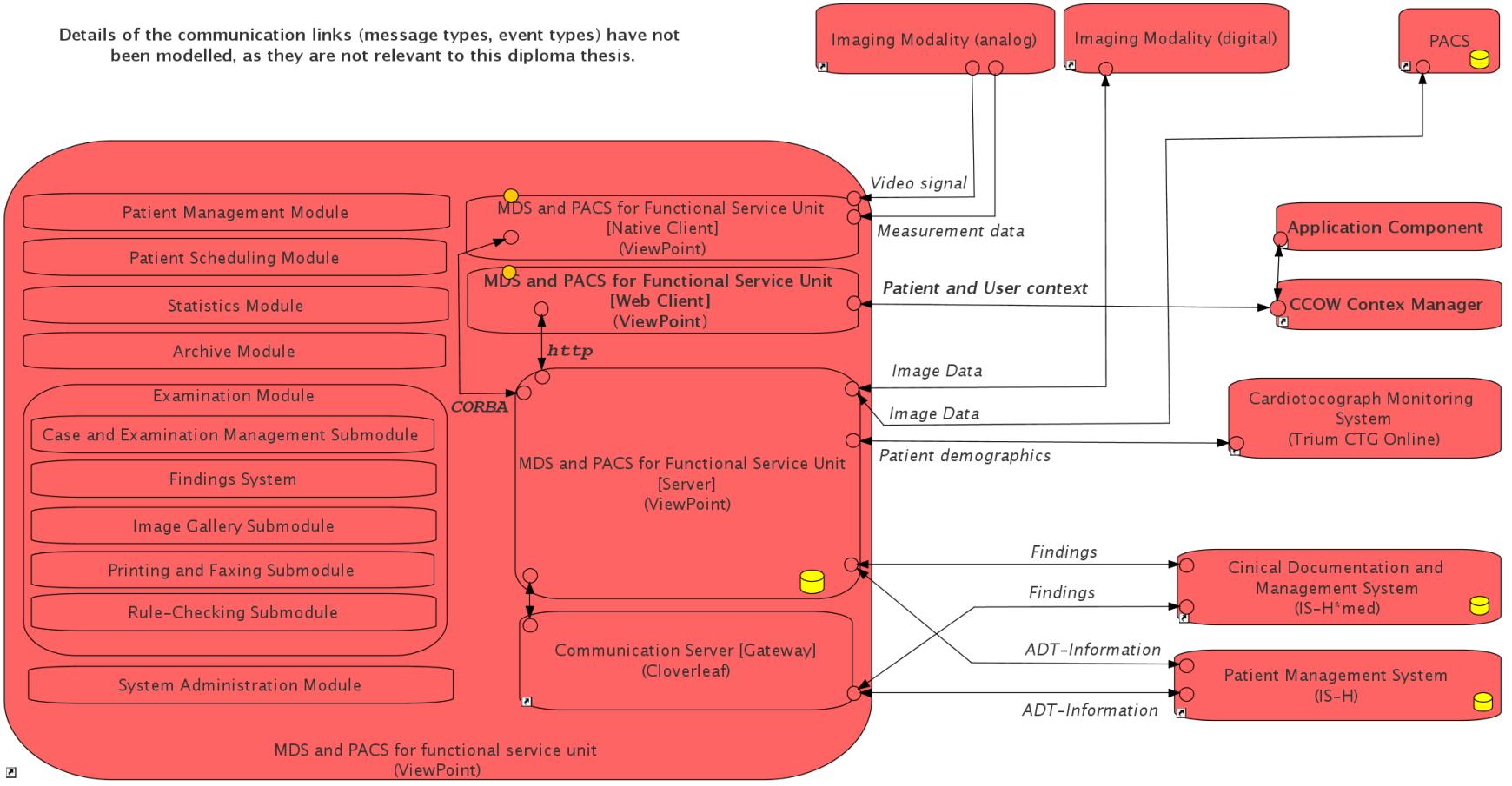


Figure A.3.: Logical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system.

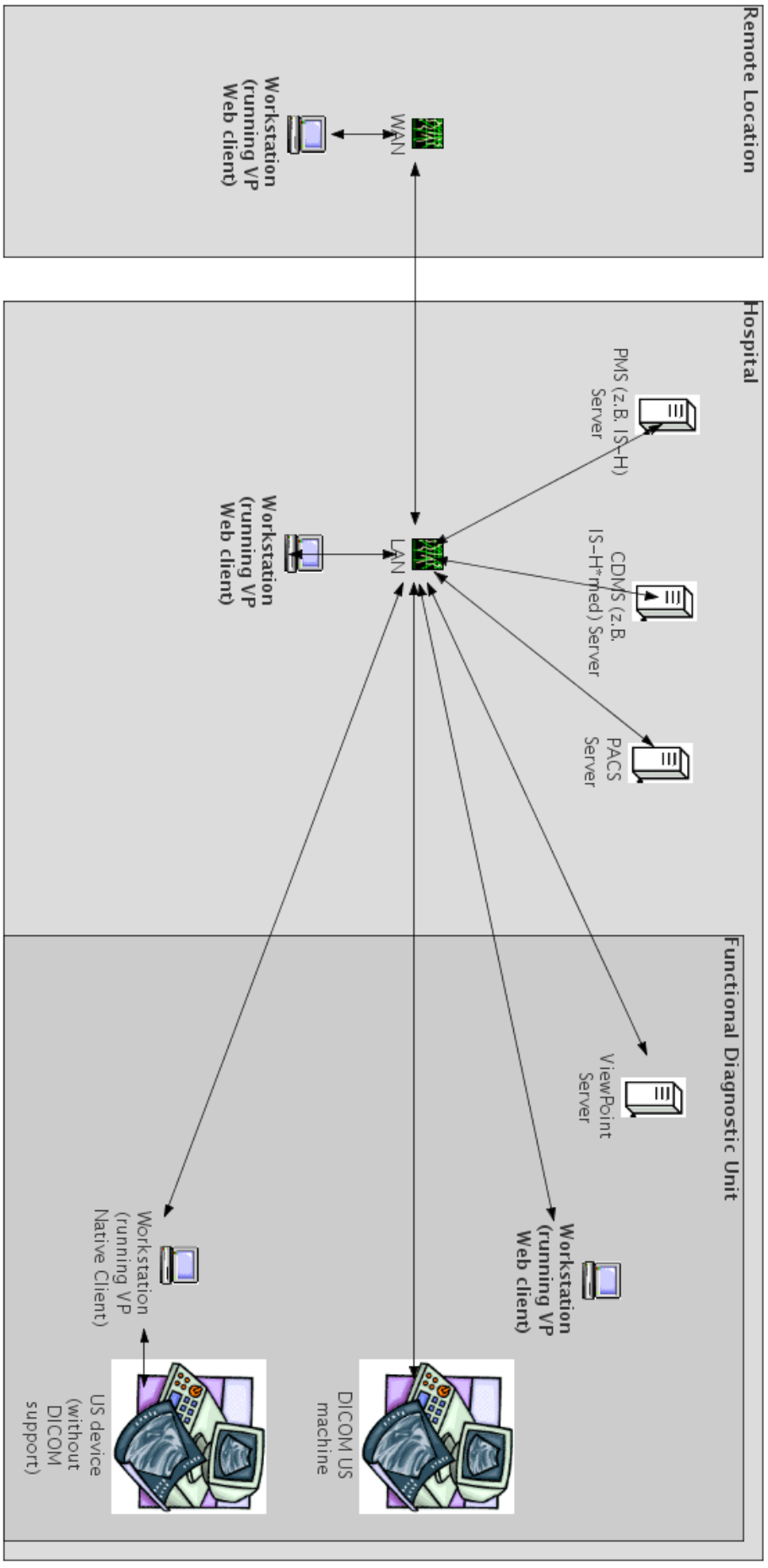


Figure A.4.: Physical tool layer of the 3LGM^2 model showing the future degree of integration of ViewPoint within the hospital information system.

Features of a web-based interface to ViewPoint

Purpose:	Version:
This questionnaire should help to determine the importance of each here listed feature of the web-based user interface of ViewPoint. Most likely the web server will be developed stepwise. Please specify in the yellow marked column, in which version the referring feature should be implemented. **Thanks a lot for your efforts!** Holger Schmuhl	**3:** Minimal Version to have as soon as possible product which is ready for the market (high importance) **2:** Next Version with new important features (medium importance) **1:** Third Version offering all intended features (low importance) **0:** not needed

Respondents:	BEH	FS	KP	WG	MN	rounded Mean
Patient Management						
Ability to search for existing patient records	3	3	3	3	3	3
Ability to manage (create, alter, delete) patient records	1	2	2	3	2	2
Patient Scheduling						
Ability to view appointments	3	3	2	2	3	3
Ability to manage appointments	1	3	2	2	2	2
History						
Ability to view the change history of a patient record and all its data	1	1	3	1	3	2
Order Management						
Ability to view orders	3	1	2	1	3	2
Ability to manage (create, alter, delete) orders	1	1	2	1	2	1
Case and Examination Management						
Ability to view the F5 text	3	3	3	2	3	3
Ability to view previously printed reports	3	1	3	2	3	2
Ability to view exam data (finding data) in the Structured Finding Masks	1	3	2	3	3	2
Ability to print referring doctor letters	3	2	3	3	3	3
Ability to alter the F5 text	2	3	2	2	1	2
Ability to alter the Structured Finding Data	1	3	2	3	2	2
Assign or unassign an order from HIS	1	1	2	3	2	2
Assign or unassign a DICOM study	1	3	2	3	2	2
Export a case to the cardiotocograph monitoring system Trium CTG Online	1	0	2	n/a in USA	3	1
Findings System						
Display Charts	2	3	3	3	3	3
Receive and record Ultrasound (US) measurement data	0	0	2	2	2	1
Image Review Capabilities						
Ability to review Single Frame Images	3	3	3	3	3	3
Ability to review Multi Frame Images (Clips)	2	3	3	3	3	3
Ability to review Raw Data (Voluson or TruAccess)	1	0	3	2	3	2
Perform measurements on images (e.g. distances on a US image)	1	2	2	2	3	2
Acquiring Image Data						
Import image data via DICOM interface	0	0	1	1	2	1
Digitize image data with the help of a Framegrabber	0	0	1	1	2	1
Digitize image data with the help of a TWAIN compatible scanner device	0	0	1	1	2	1
Import image data from file system	0	0	2	2	2	1
Import image data from clipboard	0	0	2	2	2	1

Export Image Data

Export Image data via DICOM interface	**0**	**1**	**2**	**1**	**2**	1
Export Image data to file system	**1**	**2**	**2**	**2**	**2**	2

Printing and Faxing

Ability to print	**3**	**3**	**2**	**3**	**3**	3
Ability to fax	**1**	**3**	**2**	**3**	**3**	2

Rule-checking (for perinatal findings)

Ability to check plausibility of findings	**0**	**1**	**2**	**2**	**2**	1

Additional features

Access from Intranet	**3**	**3**	**3**	**3**	**3**	3
Access from Internet	**3**	**3**	**0**	**3**	**3**	2
URL (e.g. http://viewpoint.myhospital.de/open?patid=12345)	**3**	**3**	**1**	**1**	**3**	2
Support the Clinical Context Work Group (CCOW) standard to enable contextual integration (CCOW enables contextual integration, that means when you open patient XY in application A, patient XY is automatically opened also in application B on the same workstation; CCOW additionally offers functionality to exchange login-information between different applications, so that the user needs to login only once)	**1**	**0**	**1**	**1**	**2**	1
Support HPC (Healthcare professional card is an individually programmed access card for health care professionals. This card is used to assign read and/or write access rights to specific data fields on the patient card. The chip stores the doctor's ID, read and/or write access rights, and a digital signature (for secure data transmission). In general it could be used for authorization and authentification.)	**1**	**0**	**1**	**1**	**2**	1
Enable access to knowledge servers. As major hospitals establish knowledge servers to distribute medical knowledge and guidelines hospital wide, it is conceivable that the web-based UI references to such information sources. Meaning e.g. that in case certain diagnose is selected in a patient record, a link to the referring medication and further treatment recommendations on the knowledge server will be displayed	**1**	**0**	**1**	**1**	**1**	1
Offer means for telemedical purposes, like e.g. synchronized view for the Daily Review Module	**1**	**0**	**1**	**1**	**2**	1
Support web-browsers running on Linux	**2**	**3**	**1**	**1**	**1**	2
Support web-browsers running on Mac	**1**	**1**	**0**	**1**	**1**	1
Support web-browsers running on Solaris	**0**	**0**	**0**	**1**	**1**	0
Support Mozilla Firefox Browser	**2**	**3**	**1**	**1**	**3**	2
Support Internet Explorer	**3**	**3**	**0**	**3**	**3**	2
Complete replacement of “classical” VP client by web-based UI vs. Add-on	**1**	**0**	**0**	**1**	**1**	1
Support automatic testing of the web-based UI	**3**	**2**	**3**	**1**	**2**	2

Product Analysis

Short Overview

Author/Vendor	*Euroking*
Product Name	*E3*
Product Category	*Web-based documentation system for obstetrical departements*
Goals	*Supporting in real-time the workflow*
Field of Application	*Obstrical departement*
Underlying techniques	
Web server:	*MS Internet Information Server*
Database:	*Intersystems Caché (www.intersystems.com)*
Other:	*Java Virtual Machine*
Supported Browser	*Internet Explorer 5.5 or above (with JVM activated)*
Server Platform	[X] Windows [ ] Linux/Unix [ ] Solaris
Product Type	[ ] self-tailored/unique [X] of-the-shelf
License Type	[ ] open source [X] closed source
Software Status	[ ] alpha [ ] beta [X] stable
URL	*www.euroking.com*
Contact & eMail	*John Stapleton*

Questionnaire

In case you use Apache:

1.1) Why did you choose Apache and not a closed source server (e.g. like MS Internet Information Server) or self-tailored server?

1.2) How do you handle Critical/Security Updates from apache.org?

-

In case you use another web-server:

1.1) Why did you not choose Apache instead of your current web-server?

1.2) How do you handle Critical/security Updates for your web-server?

1.3) How do you assure that your web-server is safe?

There was no big discussion, as the NHS has a special cooperation/contract with Microsoft and throughout the NHS Trusts only MS software is used. IIS was chosen and the updating/securing responsibility is in the hands of the hospitals.

With which techniques do you assure …

2.1) Secrecy/Confidentiality?

2.2) Authentication?

2.3) Nonrepudiation/Accountability?

2.4) Integrity Control?

Server is protected by firewall; user has to get on the network; until now simple login with username & password in E3; in the future usage of an integrated login (single login for all applications on a workstation, actual login procedure is carried out by MS Windows, applications just check the user name for authorization; technique will be implemented NHS wide)

3.) In which kind of environment is your system running? (e.g. local Intranet, Intranet with remote users, Internet)
Intranet; should be possible to connect clients via dial-in or VPN

4.1) Which web-browsers do you support?
Internet Browser 5.5 or above (-> NHS-Microsoft coop.)

4.2) Which adjustments/modifications has the user to make to the default browser configuration (like turning JavaScript on or installing plug-ins like Java or Macromedia Flash)?
Install/Activate Java Virtual Machine; nothing else

4.3) Are there any limitations concerning the platform on which the browser runs?
Only MS Windows (-> NHS-Microsoft coop.)

4.4) In case there is a firewall between server and client, which ports have to be open? (e.g. is it sufficient to have only port 81 for http open) Or in other words which protocols do you use?
http or https port

4.5) Which is the minimum bandwidth required for client-server communication (up-/download)?
They advise the customers to have 100 Mbps for optimal performance and real-time data entry; lower connections should be possible, they also have houses working with 10 Mbps, but as soon as the network is busy performance problems occur; one customer even got a working connection with GPRS;

4.6) To which version of XHTML, HTML, JavaScript etc. is the web interface conform?
Did not know the exact Version, standard HTML like supported by IE.

5.) Which kind of data is distributed via the web-sever? (*Text, images, videos etc.*)
Every file type can be linked in a patient record (file management system); no DICOM functionality

6.) What happens if the network is congested or the connection lost?
Major issue, as E3 is completely web-based they have no sufficient technique to prevent data loss. Their current procedure is to save/archive entered data stepwise (e.g. when another mask is opened or a field was completed) and to offer periodical auto-saving functionality

7.) Which importance do you measure to CCOW? Will you implement it in your application?
(CCOW is a standard issued by the HL7 Workgroup; CCOW enables contextual integration, that means when you open patient XY in application A, patient XY is automatically opened also in application B on the same workstation; CCOW additionally offers functionality to exchange login-information between different applications, so that the user needs to login only once)
They heard about CCOW, but they doubt that it will evolve as especially 3rd party providers will not agree that other applications "control" or access their program. Login functionality will be covered by the integrated login (single login for all applications on a workstation).

Context integration has to be done another way, e.g. by passing patient identification via a URL, which is not yet implemented.

8.) Which importance do you measure to the HealthCare Professional Card (HPC)? Will you support it in a future version of your application?
(HPC is an individually programmed access card for healthcare professionals. This card is used to assign read and/or write access to specific data fields on the patient card. The chip stores the doctor's ID, read and/or write access rights and a digital signature (for secure data transmission). In general it could be used for authorization and authentication of the application user.)
Sounds like there won't be a common standard in the UK and the card (and its content and technique) will be hospital specific, also inside the NHS; their strategy is that the authorization and authentication responsibility is at Windows, they will just retrieve the user identification from Windows and thereby determine if the user is authorized to access E3 (perhaps via ActiveX)

9.) Is it possible to work offline and to reconnect?
No, until now the old version had a special software client (not web-based!) with which offline data entry is possible (for midwifes). It synchronizes data as soon as the client has a network connection.

10.) If you would redesign and reimplement your product from scratch, what would you change?
Different DB, they are satisfied with the current one and only would do it because of popularity reasons; .NET (increase performance, better integration); concentrate more on supporting access from anywhere in the world

Interviewed via Telephone, 16.12.2005, Holger Schmuhl

Product Analysis

Short Overview

Author/Vendor	*GE Vingmed Healthcare*
Product Name	*WebEx (EchoPAC)*
Product Category	*PACS*
Goals	*High secured web architected PACS workstation, designed to handle all types of modalities.*
Field of Application	*Cardiology and other functional diagnostic units*
Underlying techniques	
Web server:	*IIS*
Database:	*Microsoft SQL Server*
Other:	*- ASP .NET* *- Media Player which is capable of playing Windows media*
Supported Browser	Internet Explorer, Netscape, Opera
Server Platform	[X] Windows [ ] Linux/Unix [ ] Solaris
Product Type	[X] self-tailored/unique [ ] of-the-shelf
License Type	[ ] open source [X] closed source
Software Status	[ ] alpha [X] beta [ ] stable
URL	-
Contact & eMail	Svein Fimreite

Questionnaire

In case you use Apache:

1.1) Why did you choose Apache and not a closed source server (e.g. like MS Internet Information Server) or self-tailored server?

1.2) How do you handle Critical/Security Updates from apache.org?
I'm not using the Apache server...

In case you use another web-server:

1.1) Why did you not choose Apache instead of your current web-server?
I prefer the IIS server, since I'm using the latest Visual Studio

1.2) How do you handle Critical/security Updates for your web-server?
I'm using the Windows Update functionality, supported by Microsoft

1.3) How do you assure that your web-server is safe?
I'm using the "test" tool in Visual Studio .NET designed for testers...

With which techniques do you assure …

2.1) Secrecy/Confidentiality?
Form security, https, 128 bit encryption; WebEx also supports auto logoff functionality and audit logging.

2.2) Authentication?
WebEx provides a custom form (Web page) for users to enter their credentials. Then a user credential token is stored in a cookie

2.3) Nonrepudiation/Accountability?
I suppose you mean how I handle the authorization which addresses the question: What can you do?WebEx has different authorization levels stored together with each user registration on the web server...

2.4) Integrity Control?
How I can guarantee that data is protected from accidental or deliberate (malicious) modification. Like privacy, integrity is a key concern, particularly for data passed across networks. WebEx uses 128 bit encryption and immediate expiry of webpages's, plus no temporary client storage. If you want to download loops and so on (on request, user dependent functionality), the content is always zipped and password protected.

3.) In which kind of environment is your system running? (e.g. local Intranet, Intranet with remote users, Internet)
It is designed for both the local intranet and the internet.

4.1) Which web-browsers do you support?
The aspx sites is converted to XHTML 1.0EN, witch means that the content is supported by all common web browsers. Its tested on Netscape, Internet Explorer, Opera and on different platforms and operating systems like MAC, LINUX and handheld devices...

4.2) Which adjustments/modifications has the user to make to the default browser configuration (like turning JavaScript on or installing plug-ins like Java or Macromedia Flash)?
I'm not using any plug-ins, so you don't need to download any extra components from the web server or to co figurate your browser. The only requirement is to accept pop-ups... Since I support full screen viewing and dual screen solutions...

4.3) Are there any limitations concerning the platform on which the browser runs?
No, but the platform must be able to display Windows Media or MPEG1 or MPEG4 loops... It can be executed outside of the browser, so the browser does not need to handle this kind of data within the browser it self...

4.4) In case there is a firewall between server and client, which ports have to be open? (e.g. is it sufficient to have only port 81 for http open) Or in other words which protocols do you use?
Port 80 - https or http (local intranet only)

4.5) Which is the minimum bandwidth required for client-server communication (up-/download)?
No minimum bandwidth. The loops will be buffered in memory - on the client side. The size of each loop depends on the frame rate and so on, but its around 1Mb in most cases. So the limitation must be set against how log time its acceptable to wait for a loop like this. The other functionalities within WebEx requires less bandwidth...

4.6) To which version of XHTML, HTML, JavaScript etc. is the web interface conform?
XHTML 1.0 and in some cases 1.1....

5.) Which kind of data is distributed via the web-sever? (*Text, images, videos etc.*)
Text, images (jpeg, gif), video(wmv(MPEG4),M1V), zip, pdf (user manual, will be converted to html)

6.) What happens if the network is congested or the connection lost?
Since the web sites expires immediately, you will get the message that its not possible to view this site in offline mode... If its in a middle of a zip download process, and the lost connection was only temporary, the downloading process will continue...

7.) Which importance do you measure to CCOW? Will you implement it in your application?
(CCOW is a standard issued by the HL7 Workgroup; CCOW enables contextual integration, that means when you open patient XY in application A, patient XY is automatically opened also in application B on the same workstation; CCOW additionally offers functionality to exchange login-information between different applications, so that the user needs to login only once)
I've not looked into this one... I know that Microsoft supports a physician digital dashboard, but I've not evaluated the benefits by supporting this functionality. I suppose it must be supported on both sides, for instance both on the EchoPAC workstation or ViewPoint and on the web client... I also suppose that this functionality will require an additional plug-in on the client browser...

8.) Which importance do you measure to the Health Care Professional Card (HPC)? Will you support it in a future version of your application?
(HPC is an individually programmed access card for healthcare professionals. This card is used to assign read and/or write access to specific data fields on the patient card. The chip stores the doctor's ID, read and/or write access rights and a digital signature (for secure data transmission). In general it could be used for authorization and authentification of the application user.)
I suppose this will require a separate program (plug-in) on each client. I'm not sure if I'm going to support HPC, it depends on the requests and how easily and flexible it can be implemented....

9.) Is it possible to work offline and to reconnect?
No...

10.) If you would redesign and reimplement your product from scratch, what would you change?
Nothing, its perfect seen from my point of view ;-)

Received via eMail, 23.12.2005, Holger Schmuhl

Product Analysis

Short Overview

Author/Vendor	*Trium Analysis Online GmbH*
Product Name	*Trium CTG Online*
Product Category	*CTG online monitoring system*
Goals	*Enable web-based CTG monitoring*
Field of Application	*mainly obstetrical departement*
Underlying techniques	
Web server:	*Apache*
Database:	*MS Access*
Other:	*- OpenSA 1.0.4 web server (including Apache SSL module and apache proxy module)* *- Macromedia ColdFusion 4.5 Express* *- National Instruments VISA* *- Adobe Acrobat Reader* *- Macromedia Flash* *- G web server*
Supported Browser	*Firefox 1.0, Mozilla 1.7, Microsoft Internet Explorer 5.5 (with JavaScript enabled, Macromedia Flash Plug-In for sound support)*
Server Platform	[X] Windows [ ] Linux/Unix [ ] Solaris
Product Type	[ ] self-tailored/unique [X] of-the-shelf
License Type	[ ] open source [X] closed source
Software Status	[ ] alpha [ ] beta [X] stable
URL	*www.trium.de*
Contact & eMail	Christian Harböck

Questionnaire

In case you use Apache:

1.1) Why did you choose Apache and not a closed source server (e.g. like MS Internet Information Server) or self-tailored server?
Apache is used as the basic communication server. We chose apache because of it is the market leader, offers several extensions (modules) and is more secure than lots of other web servers. However we added functionality using the ColdFusion application server which runs "on top" of apache.

1.2) How do you handle Critical/Security Updates from apache.org?
If updates are issued it is in the responsibility of the hospital to install them

In case you use another web-server:

1.1) Why did you not choose Apache instead of your current web-server?
1.2) How do you handle Critical/security Updates for your web-server?
1.3) How do you assure that your web-server is safe?
-

With which techniques do you assure …

2.1) Secrecy/Confidentiality?
SSL certificates

2.2) Authentication?
Username/password

2.3) Nonrepudiation/Accountability?
for alarms: user accounts

2.4) Integrity Control?
…

3.) In which kind of environment is your system running? (e.g. local Intranet, Intranet with remote users, Internet)
customer dependant; normally: local intranet but there are also customers using it remotely

4.1) Which web-browsers do you support?
See supported web browsers

4.2) Which adjustments/modifications has the user to make to the default browser configuration (like turning JavaScript on or installing plug-ins like Java or Macromedia Flash)?
SSL root CA certificate must be installed. IE requires additional configuration of security zones. Optional plugins may be installed (Adobe Reader, Macromedia Flash).

4.3) Are there any limitations concerning the platform on which the browser runs?
Only windows platforms are tested.

4.4) In case there is a firewall between server and client, which ports have to be open? (e.g. is it sufficient to have only port 81 for http open) Or in other words which protocols do you use?
HTTPS (443); 8080 for interprocess communication

4.5) Which is the minimum bandwidth required for client-server communication (up-/download)?
9600 Bauds

4.6) To which version of XHTML, HTML, JavaScript etc. is the web interface conform?
HTML 4.01; JavaScript 1.2

5.) Which kind of data is distributed via the web-sever? (*Text, images, videos etc.*)
Text, Images, Flash, PDF

6.) What happens if the network is congested or the connection lost?
We use an rotating foetus an indicator for network availability. If the foetus won't rotate, the network connection may be lost. Additionally there is a JavaScript popup window guiding the user to fix the error.

7.) Which importance do you measure to CCOW? Will you implement it in your application?
(CCOW is a standard issued by the HL7 Workgroup; CCOW enables contextual integration, that means when you open patient XY in application A, patient XY is automatically opened also in application B on the same workstation; CCOW additionally offers functionality to exchange login-information between different applications, so that the user needs to login only once)
I haven't heard about this standard before. There are currently no integration plans.

8.) Which importance do you measure to the Health Care Professional Card (HPC)? Will you support it in a future version of your application?
(HPC is an individually programmed access card for healthcare professionals. This card is used to assign read and/or write access to specific data fields on the patient card. The chip stores the doctor's ID, read and/or write access rights and a digital signature (for secure data transmission). In general it could be used for authorization and authentication of the application user.)
Starting with the German "Heilberufeausweis" this technique may get an enormous distribution. Future versions may support HPC.

9.) Is it possible to work offline and to reconnect?
No. (Wouldn't make much sense because of the type of application)

10.) If you would redesign and reimplement your product from scratch, what would you change?
Using an enterprise grade database. Replacing application server with latest version.

Received via eMail, 02.01.2006, Holger Schmuhl

B. Declaration

Hereby I affirm that I created this thesis on my own and that I used only the listed resources.

(Ich versichere, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.)

Place and date (Ort und Datum) Signature (Unterschrift)